

全国计算机技术与软件专业技术资格（水平）考试参考用书

软件设计师考试同步辅导

(上午科目)

工业和信息化部教育与考试中心 推荐

王华君 陶佳 主编 / 史国川 徐国明 副主编

第4版

清华大学出版社

全国计算机技术与软件专业技术资格(水平)考试参考用书

软件设计师考试同步辅导

(上午科目)(第4版)

全国计算机技术与软件专业技术资格(水平)考试办公室推荐

王华君　陶　佳　主　编

史国川　徐国明　副主编

清华大学出版社

北　京

内 容 简 介

本书是按照人力资源和社会保障部、工业和信息化部最新颁布的全国计算机技术与软件专业技术资格(水平)考试大纲和指定教材编写的考试用书。全书共分为12章，内容包括：计算机系统知识、程序设计语言基础、数据结构、操作系统知识、软件工程基础知识、结构化开发方法、面向对象技术、常用算法设计方法、数据库技术基础、网络与信息安全基础知识、标准化和软件知识产权基础知识、计算机专业英语，主要从考试大纲要求、考点辅导、典型例题分析、本章小结和达标训练几个方面对各部分内容加以系统地阐释。

本书具有考点分析透彻、例题典型、习题丰富等特点，非常适合备考软件设计师的考生使用，也可作为高等院校相关专业或培训班的教材。

图书在版编目(CIP)数据

软件设计师考试同步辅导(上午科目)/王华君，陶佳主编. —4版. —北京：清华大学出版社，2018
(全国计算机技术与软件专业技术资格(水平)考试参考用书)
ISBN 978-7-302-50547-1

Ⅰ. ①软…　Ⅱ. ①王…　②陶…　Ⅲ. ①软件设计—资格考试—自学参考资料　Ⅳ. ①TP311.5

中国版本图书馆CIP数据核字(2018)第145345号

责任编辑：魏　莹　李玉萍
封面设计：常雪影
责任校对：王明明
责任印制：李红英
出版发行：清华大学出版社
网　　址：http://www.tup.com.cn, http://www.wqbook.com
地　　址：北京清华大学学研大厦A座　　**邮　　编：**100084
社 总 机：010-62770175　　**邮　　购：**010-62786544
投稿与读者服务：010-62776969, c-service@tup.tsinghua.edu.cn
质量反馈：010-62772015, zhiliang@tup.tsinghua.edu.cn
印 装 者：三河市铭诚印务有限公司
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　张：**30.25　　**插　页：**4　　**字　数：**730千字
版　　次：2005年6月第1版　2018年9月第4版　　**印　次：**2018年9月第1次印刷
定　　价：89.00元

产品编号：071202-01

再 版 前 言

全国计算机技术与软件专业技术资格(水平)考试自实施起至今已经历了 20 多年，在社会上产生了很大的影响，其权威性得到社会各界的广泛认可。为适应我国信息化发展的需求，国家人力资源和社会保障部同工业和信息化部在 2009 年对软件设计师级别考试大纲进行了重新调整，以满足社会上对各种信息技术人才的需要。本书第 1 版自 2005 年、第 2 版自 2010 年、第 3 版自 2013 年出版以来，被众多考生选用为考试参考书，多次重印，深受广大读者好评。为了与考试同步，本书对第 3 版同名书进行了更新升级，将最新考试真题穿插其中。更新升级后本书特色如下。

(1) 知识点更全面。本书与 2009 年软件设计师考试大纲考试科目 1——计算机与软件工程知识基本一致，又兼顾计算机技术发展和知识更新，对属于大纲要求的知识点而指定教材没有阐述的部分进行了必要的补充。

(2) 结构与官方教程同步。本书参考最新指定官方教程、最新考试大纲及最新题型编写各章节，便于考生使用《软件设计师教程(第 5 版)》同步复习，同时更加突出重点与难点，针对性强，减轻考生复习的工作量。

(3) 例题与习题经典。2014 年至 2017 年的 8 次考试真题全部被分类解析到例题中，并同时在其中增加了根据最新考试大纲精心设计的例题，具有典型性和代表性，而 2013 年两次考试真题全部被分类归入同步练习中。使考生能从以前的考题中更好地熟悉考试的难度与广度，顺利通过考试。

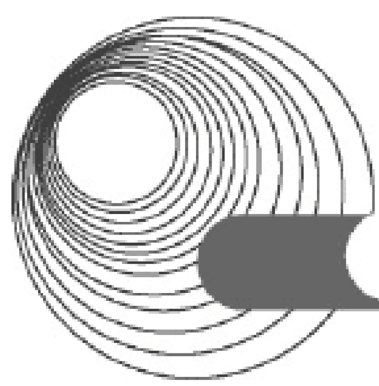
(4) 重点突出。第 4 版沿袭前一版的框架，每一小节分为 4 个模块，即考点辅导、典型例题分析、同步练习和同步练习参考答案。其中，考点辅导部分主要以专题的方式，细化软件设计师上午考试各章节的基础知识点的介绍；典型例题分析是本书的重点，它详尽细致地剖析了最近 8 次考试(2014—2017 年)的真题；同步练习每一道题都配有标准答案；每章还配有一定数量的习题及答案，可以对读者所学的知识和能力起到巩固、拓宽和提高的作用。

(5) 对语言进行了锤炼，叙述更准确，概念更清晰，覆盖所有大纲考点，并突出重点和难点。

(6) 书中所有例题与习题进行了精选，确保所有题目符合考纲要求，例题选取典型、有梯度、有广度，分析详尽；题目的难易度、分布率与真实考试相当；题目答案正确、解析科学。

本书非常适合备考软件设计师的考生使用，也可作为高等学校相关专业或培训班的教材。

本版由王华君、陶佳担任主编，史国川、徐国明担任副主编，参与本书组织、编写和资料收集的还有谢瑜、周胜、鲁磊纪、杨章静、刁爱军、陈海峰、赵晗、吴敏、刘立军、



宋白玉、石鲁生、何光明，在此对原作品作者及全体参与人员表示衷心的感谢。本章在编写的过程中，参考了许多相关的书籍和资料，从中汲取了许多营养，在此也对这些参考文献的作者表示感谢。需要特别提出感谢的是来自互联网的各位不知道姓名的网友们的无私奉献，正是由于你们，才使本书的内容更完善、更详尽。

由于作者水平所限，书中难免存在错漏和不妥之处，敬请读者批评指正。联系邮箱：iteditor@126.com。

编　者

软件设计师考试(上午)考点分布导航图

章	节	历年真题分布								大纲要求	阅读链接	命题预测
		2014.5	2014.11	2015.5	2015.11	2016.5	2016.11	2017.5	2017.11			
第 1 章 计算机系统知识	1.1 计算机系统基础知识	CPU 的组成及其部件功能(2 分)，定点数的表示(1 分)，校验码(1 分)	CPU 的组成及其部件功能(1 分)，逻辑代数(1 分)，存储器(1 分)，内存(1 分)	Cache 工作原理(1 分)，定点数和浮点数(1 分)，cache 地址映像方式	存储器(1 分)，虚拟存储器(1 分)，定点数和浮点数(1 分)	补码(1 分)，内存(1 分)	CPU 的组成及其部件功能(1 分)，程序计数器(1 分)，定点数和浮点数(1 分)，校验码(1 分)，逻辑代数(1 分)	CPU 的组成及其部件功能(1 分)，逻辑代数(1 分)，校验码(1 分)	校验码(1 分)	①二进制、十进制和十六进制等常用数制及其相互转换。 ②数据的表示，包括数的表示(原码、反码、补码、移码表示，整数和实数的机内表示，精度和溢出)，非数值表示(字符和汉字表示，声音表示，图像表示)。 ③算术运算和逻辑运算，包括计算机中的二进制数运算方法，逻辑代数的基本运算。 ④计算机系统的组成、体系结构分类及特性，包括 CPU 和存储器的组成、性能和基本工作原理，常用 I/O 设备、通信设备的性能，以及基本工作原理，I/O 接口的功能、类型和特性；CISC/RISC，流水线操作，多处理机，并行处理。 ⑤存储系统，包括虚拟存储器基本工作原理，多级存储体系的性能和价格，RAID 的类型和特性。 ⑥安全性、可靠性与系统性能评测基础知识，包括系统可靠性分析评价和计算机系统性能评测方式。 ⑦加密和解密机制	**阅读建议** 本章对应《软件设计师教程(第 5 版)，清华大学出版社出版》(以下简称“教程”)第 1 章“计算机系统知识”。章节的结构安排与教程完全同步，考生可以对照教程进行同步复习	本章节考点分值约占总考分的 9%，每年所占比分差别不大。通常考查数据表示，计算机系统的组成，存储系统。高考频的考点为： ●浮点数的运算； ●校验码； ●CPU 和存储器的组成及其部件功能； ●流水线操作； ●寻址方法； ●Cache； ●虚拟存储器； ●指令的执行方式； ●CISC/RISC； ●吞吐率
	1.2 计算机体系结构	指令分析过程(2 分)	CISC 和 RISC 的区别(1 分)	系统总线(1 分)，中断(1 分)，指令流水线(1 分)	寻址方式(1 分)，内存编址(1 分)，CISC 和 RISC 的区别(1 分)	Cache 与主存调度(1 分)，指令分析过程(2 分)，总线结构作用(1 分)，中断(1 分)	指令流水线(1 分)，Cache (1 分)	输入输出方式(1 分)，Cache (1 分)	中断(1 分)，Cache (1 分)，指令分析过程(2 分)，内存编址(1 分)，内存(1 分)			
	1.3 安全性、可靠性与系统评测基础知识	防火墙(2 分)，病毒(1 分)	防火墙(1 分)，病毒(1 分)，安全协议(1 分)		防火墙(1 分)		数字签名(2 分)	数字签名(1 分)，可靠度计算(1 分)，加密算法(1 分)	防火墙(1 分)			

续表

章	节	历年真题分布								大纲要求	阅读链接	命题预测
		2014.5	2014.11	2015.5	2015.11	2016.5	2016.11	2017.5	2017.11			
第2章程序设计语言基础	2.1 程序设计基础知识	程序语言特点(1分)，参数传递方式(1分)，函数调用(1分)，编译的概念(1分)，编译与解释(1分)	中间代码的概念(1分)，编译的概念(1分)	程序语言特点(1分)，变量与常量(1分)，编译与解释(1分)，NFA和DFA(1分)	编译与解释(2分)，NFA和DFA(1分)，函数调用(1分)	参数调传递方式(1分)，编译与解释(1分)，脚本语言(1分)，中间代码的概念(1分)，NFA和DFA(1分)	参数传递方式(1分)，程序的控制结构(1分)	程序语言特点(1分)，DFA(1分)，参数传递方式(1分)，字符串(1分)	程序的概念(1分)，程序语言特点(2分)，编译与解释(2分)，编译和语法分析(1分)，NFA和DFA(1分)	①汇编、编译、解释系统的基础知识和基本工作原理。②程序设计语言的基本成分(数据、运算、控制和传输)，程序调用的实现机制。③各类程序设计语言的主要特点和适用情况	**阅读建议** 本章对应教程第2章“程序语言基础知识”，章节的结构安排与教程完全同步。考生可以对照教程进行同步复习	本章节考点分值约占总考分的5%，其比例较前几年有所下降，不同年份所占分值比例又稍有不同。通常考查程序调用的实现机制，各类程序设计语言的特点。高考频的考点为： ●参数传递方式； ●程序设计语言的基本成分； ●DFA、NFA； ●文法和正规式； ●汇编、编译、解释系统的基础知识和基本原理； ●各类程序设计语言主要特点
	2.2 语言处理程序基础	编译和语法分析(2分)	有限自动机(1分)，语法描述(1分)	语法分析法(1分)	表达式(1分)	编译和语法分析(2分)	正规表达式(1分)，语法描述(1分)	正规表达式(1分)，编译和语法分析(1分)，后缀表达式(1分)	表达式(1分)			
第3章数据结构	3.1 线性结构	顺序表(1分)，队列的性质(1分)	栈的作用(1分)	循环队列头指针(1分)，栈和队列区别(1分)，队列(2分)	栈和队列区别(1分)，	栈的作用(1分)	顺序表(1分)，单向循环链表特征(1分)	栈的用法(1分)		①数组的定义、存储、操作。 ②链表的定义、存储、操作。 ③队列的定义、存储、操作。 ④栈的定义、存储、操作。 ⑤树的定义、存储、操作。 ⑥图的定义、存储、操作。 ⑦Hash表	**阅读建议** 本章对应教程第3章“数据结构”。章节的结构安排与教程完全同步。考生可以对照教程进行同步复习	本章节考点分值约占总考分的9%，不同年份所占分值比例又稍有不同。通常考查线性结构，树和图的相关知识，排序的相关计算。高考频的考点为： ●线性表的存储结构； ●链表(单向链表、双向链表、循环链表)的定义、存储和操作； ●队列、栈、树(二叉树、查找树、平衡树、线索树、线索树、堆)的定义、存储和操作； ●图的定义、存储和操作； ●几种常见排序方法及其时间复杂度
	3.2 数组、矩阵和广义表				三角矩阵存储位置(1分)							
	8.3 树	二叉树的遍历(2分)	二叉树的性质(1分)	二叉树的性质(1分)	二叉树的遍历(1分)	二叉树的遍历(1分)，二叉树的性质(1分)	二叉树的性质(2分)	二叉树遍历(1分)	哈夫曼树的构造(1分)，二叉树的性质(1分)			
	3.4 图				图的存储的基本概念(1分)	图的遍历(1分)	识别拓扑排序(1分)		无向图的概念(1分)			
	3.5 查找	二分查找(1分)		折半查找(1分)	折半查找(1分)	二分查找(1分)		二分查找(1分)				
	3.6 排序			简单排序(1分)，快速排序(1分)，算法时间复杂度(1分)					归并排序次数(1分)，快速排序(1分)，算法时间复杂度(2分)			

续表

章	节	历年真题分布								大纲要求	阅读链接	命题预测
		2014.5	2014.11	2015.5	2015.11	2016.5	2016.11	2017.5	2017.11			
第 4 章操作系统知识	4.1 操作系统基础知识	操作系统的概念与组成（1分）					实时操作系统(1 分)			①操作系统的内核(中断控制)、进程、线程概念。 ②处理机管理(状态转换、共享与互斥、分时轮转、抢占、死锁)。 ③存储管理(主存保护、动态链接分配、分段、分页、虚存)。 ④设备管理(I/O 控制、假脱机)。 ⑤文件管理(文件目录、文件组织、存取方法、存取控制、恢复处理)。 ⑥作业管理(作业调度、作业控制语言(JCL)、多道程序设计)。 ⑦网络操作系统和嵌入式操作系统的基础知识。 ⑧操作系统的配置	**阅读建议** 本章对应教程第 4 章“操作系统知识”。章节的结构安排与教程完全同步。考生可以对照教程进行同步复习	本章节考点分值约占总考分的10%，不同年份所占分值比例又稍有不同。通常考查处理机管理、存储管理和文件管理。高考频的考点为： ●PV 操作； ●安全序列和死锁； ●进程的同步与互斥； ●磁盘调度算法； ●地址变换的相关计算。 ●操作系统的基本概念； ●状态转换图
	4.2 进程管理	死锁（1分)，进程（1 分）	进程资源图(2 分)	PV 操作(3 分)，	PV 操作（2分），死锁(1分)，线程（1分）	PV 操作(3 分)	PV 操作(1 分)	死锁(1 分)，PV 操作(3 分)	进程资源图(2 分)			
	4.3 存储管理		页式存储(1 分)	页面变换(1 分)	缓冲技术(2 分)		页式存储(1 分)	页面变换(1 分)	页式存储(1 分)			
	4.4 设备管理		磁盘扫描时间(2 分)			磁盘扫描时间(1 分)						
	4.5 文件管理	文件路径(2 分)				文件路径(1 分)		Linux 文件路径(1 分)	文件地址索引(1 分)			
	4.6 作业管理											

续表

章	节	历年真题分布								大纲要求	阅读链接	命题预测
		2014.5	2014.11	2015.5	2015.11	2016.5	2016.11	2017.5	2017.11			
第5章软件工程基础知识	5.1 软件工程基础知识	软件开发模型分类(1分)，成本估算(1分)，pert图和gantt图(1分)，风险分析(1分)，开发模型(2分)，冗余技术(1分)，McCabe度量(1分)，软件文档(1分)，软件可维护性(1分)	软件开发模型分类(1分)，风险分析(1分)，CMM模型(1分)，ISO/IEC 9126软件质量模型(1分)，软件文档(1分)，McCabe度量(1分)，软件可维护性(1分)	软件工程基本要素(1分)，软件生命周期(1分)，软件开发模型分类(1分)，敏捷开发(1分)，McCabe度量(1分)，风险控制(1分)，配置管理(1分)	开发模型(4分)，项目进度管理(2)分，配置管理(1分)，ISO/IEC 9126软件质量模型(1分)，数据流图(1分)	项目进度管理(2分)，需求分析(1分)，增量开发模型(1分)，软件可维护性(1分)，McCabe度量(1分)	软件开发模型(1分)，极限编程(1分)，项目进度管理(2分)，估算(1分)，瀑布模型(2分)，McCabe度量(1分)，ISO/IEC 9126软件质量模型(1分)	项目进度管理(2分)，沟通路径(1分)，螺旋模型(1分)，极限编程(1分)，McCabe度量(2分)	项目进度管理(2分)，需求分析(1分)，配置管理(1分)，极限编程(1分)	①软件工程知识，包括软件开发方法，软件开发项目管理基础知识，风险管理(风险分析、风险类型、抗风险措施和内部控制)，软件开发工具与软件开发环境，统一过程(UP)与极限编程(XP)的基本概念。 ②软件质量特性(ISO/IEC 9126软件质量模型)和软件质量保证，软件评审(设计质量评审、程序质量评审)。 ③软件复杂性的概念和度量方法(McCabe度量法)。 ④诊断与容错技术。 ⑤软件能力成熟度模型CMM。 ⑥系统分析基础知识，包括结构化分析方法(数据流图、数据字典、实体关系图、描述加工处理的结构化语言)，统一建模语言，系统规格说明书。 ⑦系统设计知识，包括结构化设计方法，系统详细设计，系统设计说明书，系统分析的主要步骤，系统设计的基本原理，概要设计与详细设计的基本任务。 ⑧系统实施知识，包括结构化程序设计、面向对象程序设计，程序设计方法，系统测试，系统转换。 ⑨系统运行和维护知识，包括系统可维护性知识，系统评价基础知识，软件生存周期与软件生存周期模型	**阅读建议** 本章对应教程第5章。章节的结构安排与教程完全同步。考生可以对照教程进行同步复习	本章节考点分值约占总考分的17%，考生应充分重视本章的复习。通常考查软件工程基础知识，系统设计和实施知识。高考频的考点为： ●CMM模型； ●ISO/IEC 9126软件质量模型； ●软件开发过程管理； ●风险分析； ●PERT图和甘特图； ●白盒测试和黑盒测试的方法； ●结构化开发方法； ●系统文档相关知识； ●耦合和内聚的概念； ●测试方法分类； ●软件维护的基本知识
	5.2 系统分析基础知识				结构化分析(1分)	结构化分析(2分)		数据流图(1分)				
	5.3 系统设计基础知识			模块耦合类型(1分)，质量评审(1分)		模块耦合类型(1分)	模块内聚类型(1分)	模块调用关系(1分)，软件体系结构(1分)，模块设计原则(1分)，模块内聚类型(2分)	模块耦合类型(1分)，模块内聚类型(1分)			
	5.4 系统实施知识	白盒测试(2分)	测试基本方法(1分)		白盒测试(2分)	集成测试(2分)	测试(1分)，白盒测试(1分)		白盒测试(1分)，设计测试用例(1分)			
	5.5 系统运行和维护知识			软件维护的基本知识(1分)	软件维护的基本知识(1分)		软件维护的基本知识(1分)		软件维护(1分)，维护的类型(1分)			

续表

章	节	历年真题分布								大纲要求	阅读链接	命题预测
		2014.5	2014.11	2015.5	2015.11	2016.5	2016.11	2017.5	2017.11			
第 6 章结构化开发方法	6.1 系统分析与设计概述				结构化分析(1分)	结构化分析(2分)		数据流图(1 分)		本章内容是第 5 版教程新独立出来的章节，并丰富了内容。部分内容可以借鉴上章内容的大纲要求	**阅读建议** 本章对应教程第 6 章章节的结构安排与教程完全同步。考生可以对照教程进行同步复习	本章节考点分值约占总考分的 6%，不同年份所占分值比例又稍有不同。通常考查网络互连硬件的基础知识，Internet 及应用的相关知识，网络安全知识。高考频的考点为： ●常用的几个网络互连硬件； ●TCP/IP 协议； ●电子邮件协议； ●IP 地址分类； ●ping 命令； ●网络攻击； ●网络设计原则； ●防火墙的基本概念
	6.2 结构化分析方法	相关内容没有出现过考题，但是大纲上依然明确要求掌握本节内容。										
	6.3 结构化设计方法			模块耦合类型(1 分)，质量评审(1分)		模块耦合类型(1 分)	模块内聚类型(1分)	模块调用关系(1分)，软件体系结构(1 分)，模块设计原则(1 分)，模块内聚类型(2 分)	模块耦合类型(1 分)，模块内聚类型(1 分)			
	6.4 WebApp 分析与设计	相关内容没有出现过考题，本节内容主要是概念性的，是本章的基础，建议考生掌握										
	6.5 用户界面设计	相关内容没有出现过考题，但建议考生掌握										
第 7 章面向对象技术	7.1 面向对象的基本概念	类和对象的概念(1 分)，继承、多态、消息和封装的概念(2 分)，绑定的概念(3 分)	封装、继承、多态和绑定的概念(3 分)，类和对象的概念(1 分)	封装、继承、多态和绑定的概念(1 分)，重载的概念(1 分)	类和对象的概念(2 分)	类和对象的概念(2 分)	动态绑定(1 分)，多态(1 分)，	继承(2 分)，多态(1 分)，对象(1 分)，消息(1 分)	封装、继承、多态和绑定的概念(2 分)，对象的特性(1 分)	①面向对象的基本概念。 ②面向对象分析与设计方法。 ③面向对象程序设计知识。 ④面向对象数据库、分布式对象的概念。 ⑤分析模式与设计模式知识	**阅读建议** 本章对应教程第 7 章“面向对象技术”。章节的结构安排与教程完全同步。考生可以对照教程进行同步复习	本章节考点分值约占总考分的 9%，不同年份所占分值比例又稍有不同。通常考查面向对象的基本概念，UML 相关知识，设计模式。高考频的考点为： ●继承、多态、重置和封装的概念； ●面向对象分析的概念； ●UML 类图； ●几种设计模式。 ●面向对象的相关基本概念
	7.2 面向对象程序设计			面向对象设计的描述(1 分)								
	7.3 面向对象开发技术		面向对象分析的概念(1 分)				面向对象分析的描述 (1 分)	面向对象分析的描述 (1 分)				
	7.4 面向对象的分析与设计方法	UML 中各种图的功能(1 分)，UML 序列图(2 分)	UML 部署图(1 分)，UML 中关联关系(2 分)	UML 中关联关系(2 分)，UML 中各种图的功能(2 分)	类图(1 分)，用例图(3 分)，组件图(1 分)	UML 中各种图的功能(3 分)，用例图(1 分)，UML 关联多重度(1 分)	UML 状态图(2 分)，类图(2 分)		类图(3 分)			
	7.5 设计模式	适配器模式(1 分)，装饰器(2 分)，设计模式一些相关概念(1 分)	设计模式一些相关概念(2 分)	组合模式(3 分)，设计模式一些相关概念(1 分)	单例模式(1 分)，装饰器(1 分)，观察者模式(1 分)，工厂方法模式(1 分)	抽象工厂模式(1 分)，观察者模式(1 分)，命令模式(1 分)，享元模式(1 分)	生成器模式(2 分)	观察者模式(2 分)，装饰器(1 分)，外观模式(1 分)	状态模式(3 分)			

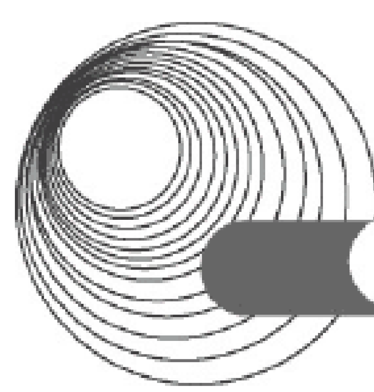
续表

章	节	历年真题分布								大纲要求	阅读链接	命题预测
		2014.5	2014.11	2015.5	2015.11	2016.5	2016.11	2017.5	2017.11			
第8章常用算法设计方法	8.1 算法和算法设计的基本概念		算法设计策略(1分)				算法设计策略(2分)			①排序算法、查找算法、数值计算方法、字符串处理方法、数据压缩算法、递归算法、图的相关算法。 ②算法描述和分析。 ③图的相关算法	**阅读建议** 本章对应教程第8章“算法分析与设计”。章节的结构安排与教程完全同步。考生可以对照教程进行同步复习	本章节考点分值约占总考分的4%，不同年份所占分值比例基本相同。通常考查算法分析基础，分治法和贪心法。高考频的考点为： ●算法时间复杂度的计算； ●各种典型算法的设计思想； ●算法的五个特性
	8.2 算法分析基础	时间复杂度的计算(2分)	时间复杂度的计算(1分)		算法的复杂度(1分)，算法的时间复杂度计算(1分)	算法的时间复杂度计算(2分)	时间复杂度的计算(1分)	时间复杂度的计算(1分)				
	8.3 分治法											
	8.4 动态规划法					动态规划寻找最优解(2分)	动态规划寻找最优解(2分)	动态规划寻找最优解(3分)				
	8.5 贪心法	贪心法的设计思想(1分)										
	8.6 回溯法											
	8.7 分支限界法	相关内容没有出现过考题，但是大纲上依然明确要求掌握本节内容										
	8.8 概率算法	相关内容没有出现过考题，但是大纲上依然明确要求掌握本节内容										
	8.9 近似算法	相关内容没有出现过考题，但是大纲上依然明确要求掌握本节内容										
	8.10 数据挖掘算法	相关内容没有出现过考题，但是大纲上依然明确要求掌握本节内容										
	8.11 智能优化算法	相关内容没有出现过考题，但是大纲上依然明确要求掌握本节内容										
第9章数据库技术基础	9.1 基本概念	软件恢复的概念(2分)	数据库设计基础(2分)		数据库设计基础(2分)，分布式数据库(2分)	数据库设计基础(2分)	数据库设计基础(1分)			①数据库管理系统的功能和特征。 ②数据库模型(概念模式、外模式、内模式)。 ③数据模型，E-R图，规范化。 ④数据操作。 ⑤数据库语言(SQL)。 ⑥数据库的控制功能。 ⑦数据仓库和分布式数据库基础知识	**阅读建议** 本章对应教程第9章“数据库技术基础”。章节的结构安排与教程完全同步。考生可以对照教程进行同步复习	本章节考点分值约占总考分的10%，不同年份所占分值比例又稍有不同。通常考查数据模型，关系代数，关系数据库和SQL语言。高考频的考点为： ●数据库的三级模式； ●数据库的安全控制机制； ●关系模式的相关知识； ●关系代数的运算； ●SQL语句； ●数据库的并发控制； ●范式的概念以及各个范式之间的区别
	9.2 数据模型							联系类型及关系模式(1分)	联系类型及关系模式(7分)			
	9.3 关系代数		关系代数的运算(1分)	关系代数的运算(1分)			关系代数的运算(3分)	关系代数的运算1分)				
	9.4 关系数据库SQL简介	SQL语句(3分)	SQL语句(3分)					SQL语句(2分)	JDBC(1分)			
	9.5 关系数据库规范化			关系模式(4分)	传递依赖(2分)，	关系模式(4分)	关系模式(2分)					
	9.6 数据库的控制功能											

续表

章	节	历年真题分布								大纲要求	阅读链接	命题预测
		2014.5	2014.11	2015.5	2015.11	2016.5	2016.11	2017.5	2017.11			
第10章网络与信息安全基础知识	10.1 网络概述	相关内容没有出现过考题，本节内容主要是概念性的，是本章的基础，建议考生掌握。								①网络体系结构(网络拓扑，OSI/RM，基本的网络协议)。 ②传输介质、传输技术、传输方法、传输控制。 ③常用网络设备和各类通信设备。 ④Client/Server 结构、Browser/Server 结构。 ⑤LAN 拓扑，存取控制，LAN 的组网，LAN 间连接，LAN-WAN 连接。 ⑥因特网基础知识以及应用。 ⑦网络软件。 ⑧网络管理、网络性能分析	**阅读建议** 本章对应教程第 10 章“网络基础知识”。章节的结构安排与教程完全同步。考生可以对照教程进行同步复习	本章节考点分值约占总考分的 6%，不同年份所占分值比例又稍有不同。通常考查网络互连硬件的基础知识，Internet 及应用的相关知识，网络安全知识。高考频的考点为： ●常用的几个网络互连硬件； ●TCP/IP 协议； ●电子邮件协议； ●IP 地址分类； ●ping 命令； ●网络攻击； ●网络设计原则； ●防火墙的基本概念
	10.2 ISO/OSI 网络体系结构	相关内容没有出现过考题，但是大纲上依然明确要求掌握本节内容。										
	10.3 网络互联的硬件											
	10.4 网络的协议与标准											
	10.5 Internet 及应用	IP 地址分类(1 分)，地址段划分(1 分)，网络访问（2 分）						IP 地址分类(1 分)，地址段划分(1 分)，网络访问（2 分）				
	10.6 网络安全											
第11章标准化和软件知识产权基础知识	11.1 标准化基础知识									①标准化意识，标准化组织机构，标准的内容、分类、代号与编号规定，标准制订过程。 ②国际标准、国家标准、行业标准、企业标准。 ③代码标准、文件格式标准、安全标准、软件开发规范和文档标准。 ④全球信息化趋势、国家信息化战略、企业信息化战略和策略。 ⑤远程教育、电子商务、电子政务等基础。 ⑥企业信息资源管理基础知识。 ⑦互联网相关标准、基于构件的软件标准。 ⑧互联网相关的法律、法规，保护知识产权有关的法律、法规。 ⑨个人信息保护规则	**阅读建议** 本章对应教程第 11 章“标准化和软件知识产权基础知识”。章节的结构安排与教程完全同步。考生可以对照教程进行同步复习	本章节考点分值约占总考分的 3%，2006 年到 2009 年的 8 次考试中该部分内容都占 2 分。通常考查标准化相关概念，侵权问题，知识产权的维护。高考频的考点为： ●标准化相关概念； ●基本标准代号格式； ●侵权问题； ●软件著作权问题； ●专利申请
	11.2 知识产权基础知识	专利权(1 分)，软件经济权利的许可使用(1 分)，通信标准(1 分)	著作权(1 分)，商标权归属(1 分)	专利权(1 分)，著作权(1 分)	著作权和专利权归属(1 分)，著作权(1 分)	著作权和专利权归属(1 分)，专利权归属(1 分)	专利权归属(1 分)，商标权 (2 分)	专利权归属(1 分)，软件著作权的知识(1 分)，商标(1 分)	著作权(1 分)，商标权(2 分)，专利的享有权(1 分)			
※第12章计算机专业英语	12.1 专业英语试题分析	阅读理解填空(5 分)	阅读理解填空(5 分)	阅读理解填空(5 分)	阅读理解填空(5 分)	阅读理解填空(5 分)	阅读理解填空(5 分)	阅读理解填空(5 分)	阅读理解填空(5 分)	①具有工程师所要求的英语阅读水平。 ②理解本领域的英语术语	**1. 阅读建议** 本章节内容教程上没有相关内容。考生可完全参看本书复习。 **2. 贴心提醒** 本章节内容大纲上明确提出，故本书将其整理编排，补充为第 12 章计算机专业英语	本章节考点分值每年固定为 5 分，考查范围比较广泛，需要考生广泛阅读。高频率的考点为： ●计算机英语基本词汇； ●阅读简单的英语资料

目　录



第 1 章　计算机系统知识

大纲要求：

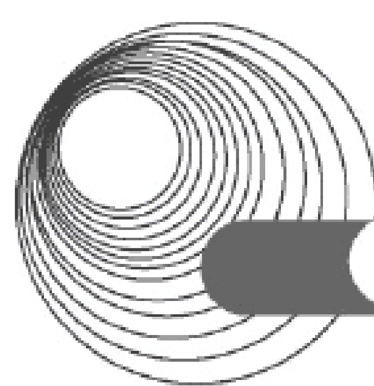
- 数值及其转换，包括二进制、十进制和十六进制等常用数制及其相互转换。
- 数据的表示，包括数的表示(原码、反码、补码、移码表示，整数和实数的机内表示，精度和溢出)、非数值表示(字符和汉字的表示、声音的表示、图像的表示)、校验方法和校验码。
- 算术运算和逻辑运算，包括计算机中的二进制数运算方法、逻辑代数的基本运算和逻辑表达式的化简。
- 计算机系统的组成、体系结构的分类及特性，包括 CPU 和存储器的组成、性能和基本工作原理，常用 I/O 设备、通信设备的性能以及基本工作原理，I/O 接口的功能、类型和特性，I/O 控制方式(中断系统、DMA、I/O 处理机方式)，CISC/RISC，流水线操作，多处理机，并行处理。
- 存储系统，包括虚拟存储器的基本工作原理、多级存储体系的性能价格、RAID 的类型和特性。
- 加密与解密机制。
- 安全性、可靠性与系统性能评测的基础知识，包括系统可靠性分析评价和计算机系统性能评测方式。
- 系统性能知识，包括性能指标(响应时间、吞吐量、周转时间)和性能设计、性能测试和性能评估。

1.1　计算机系统基础知识

1.1.1　考点辅导

1.1.1.1　计算机系统硬件的基本组成

计算机的基本硬件系统由运算器、控制器、存储器、输入设备和输出设备五大部件组成。运算器、控制器等部件被集成在一起，统称为中央处理单元(Central Processing Unit，CPU)。CPU 是硬件系统的核心，用于数据的加工处理，能完成各种算术、逻辑运算及控制功能。存储器是计算机系统中的记忆设备，分为内部存储器和外部存储器。前者速度高、容量小，一般用于临时存放程序、数据及中间结果。后者容量大、速度慢，可以长期保存程序和数据。输入设备和输出设备合称为外部设备(简称外设)，输入设备用于输入原始数据及各种命令，输出设备则用于输出计算机运行的结果。



1.1.1.2　中央处理单元

1. CPU 的功能

CPU 具有以下功能。

(1) 程序控制。CPU 通过执行指令来控制程序的执行顺序，这是 CPU 的重要职能。

(2) 操作控制。一条指令功能的实现需要若干操作信号来完成，CPU 产生每条指令的操作信号并将操作信号送往不同的部件，控制相应的部件按指令的功能要求进行操作。

(3) 时间控制。CPU 对各种操作进行时间上的控制，这就是时间控制。CPU 对每条指令的整个执行时间要进行严格控制。同时，指令执行过程中操作信号的出现时间、持续时间及出现的时间顺序都需要进行严格控制。

(4) 数据处理。CPU 通过对数据进行算术运算及逻辑运算等方式进行加工处理，数据加工处理的结果被人们所利用。所以，对数据的加工处理是 CPU 最根本的任务。

2. CPU 的组成

CPU 主要由运算器、控制器、寄存器组和内部总线等部件组成。

1) 运算器

运算器由算术逻辑单元(Arithmetic and Logic Unit，ALU)、累加寄存器、数据缓冲寄存器和状态条件寄存器组成，它是数据加工处理部件，完成计算机的各种算术和逻辑运算。相对控制器而言，运算器接受控制器的命令而进行动作，即运算器所进行的全部操作都是由控制器发出的控制信号来指挥的，所以它是执行部件。运算器有以下两个主要功能。

(1) 执行所有的算术运算，如加、减、乘、除等基本运算及附加运算。

(2) 执行所有的逻辑运算并进行逻辑测试，如与、或、非、零值测试或两个值的比较等。

下面简要介绍运算器各部件的组成和功能。

(1) 算术逻辑单元。ALU 是运算器的重要组成部件，负责处理数据，实现对数据的算术运算和逻辑运算。

(2) 累加寄存器(AC)。AC 通常简称为累加器，它是一个通用寄存器。其功能是当运算器的算术逻辑单元执行算术或逻辑运算时，为 ALU 提供一个工作区。

(3) 数据缓冲寄存器(DR)。在对内存储器进行读写操作时，用 DR 暂时存放由内存储器读写的一条指令或一个数据字，并将不同时间段内读写的数据隔离开来。

(4) 程序状态字寄存器(PSW)。PSW 保存由算术指令和逻辑指令运行或测试的结果建立的各种条件码和内容，主要分为状态标志和控制标志，如运算结果进位标志(C)、运算结果溢出标志(V)、运算结果为 0 标志(Z)、运算结果为负标志(N)、中断标志(I)、方向标志(D)和单步标志等。这些标志通常分别由一位触发器保存，保存了当前指令执行完成之后的状态。通常，一个算术操作产生一个运算结果，而一个逻辑操作则产生一个判决。

2) 控制器

运算器只能完成运算，而控制器用于控制整个 CPU 的工作，它决定了计算机运行过程的自动化。它不仅要保证程序的正确执行，而且要能够处理异常事件。控制器一般包括指令控制逻辑、时序控制逻辑、总线控制逻辑和中断控制逻辑等几个部分。

(1) 指令寄存器(IR)。当 CPU 执行一条指令时，先把它从内存储器取到缓冲寄存器中，

再送入 IR 暂存，指令译码器根据 IR 的内容产生各种微操作指令，控制其他的组成部件工作，从而完成所需的功能。

(2) 程序计数器(PC)。PC 具有寄存信息和计数两种功能，又称为指令计数器。程序的执行分为两种情况，一种是顺序执行，另一种是转移执行。在程序开始执行前，将程序的起始地址送入 PC，该地址在程序加载到内存时确定，因此 PC 的内容即是程序第一条指令的地址。执行指令时，CPU 将自动修改 PC 的内容，以便使其保存的总是将要执行的下一条指令的地址。由于大多数指令都是按顺序来执行的，所以修改的过程通常只是简单地对 PC 加 1。当遇到转移指令时，后继指令的地址根据当前指令的地址加上一个向前或向后转移的位移量得到，或者根据转移指令给出的直接转移的地址得到。

(3) 地址寄存器(AR)。AR 保存当前 CPU 所访问的内存单元的地址。由于内存和 CPU 存在操作速度上的差异，所以需要使用 AR 保存地址信息，直到内存的读/写操作完成为止。

(4) 指令译码器(ID)。指令分为操作码和地址码两部分，为了能执行任何给定的指令，必须对操作码进行分析，以便识别所完成的操作。指令译码器就是对指令中的操作码字段进行分析和解释，识别该指令规定的操作，向操作控制器发出具体的控制信号，控制各部件工作，从而完成所需的功能。

3) 寄存器组

寄存器组可分为专用寄存器和通用寄存器。运算器和控制器中的寄存器是专用寄存器，其作用是固定的。通用寄存器用途广泛并可由程序员规定其用途，其数目因处理器不同而有所差异。

4) 内部总线

内部总线(Internal Bus)将处理器的所有结构单元内部相连。它的宽度可以是 8 位、16 位、32 位或 64 位。

目前比较流行的内部总线技术有以下几种。

(1) I^2C 总线。

I^2C(Inter-Integrated Circuit，内部集成电路)总线 10 多年前由 Philips 公司推出，是近年来在微电子通信控制领域广泛采用的一种新型总线标准。它是同步通信的一种特殊形式，具有接口线少、控制方式简化、器件封装体积小、通信速率较高等优点。在主从通信中，可以有多个 I^2C 总线器件同时接到 I^2C 总线上，通过地址来识别通信对象。

(2) SPI 总线。

串行外围设备接口(Serial Peripheral Interface，SPI)总线技术是 Motorola(摩托罗拉)公司推出的一种同步串行接口。Motorola 公司生产的绝大多数 MCU(微控制器)都配有 SPI 硬件接口，如 68 系列 MCU。SPI 总线是一种三线同步总线，因其硬件功能很强，所以，与 SPI 有关的软件就相当简单，使 CPU 有更多的时间处理其他事务。

(3) SCI 总线。

串行通信接口(Serial Communication Interface，SCI)也是由 Motorola 公司推出的。它是一种通用异步通信接口 UART，与 MCS-51 的异步通信功能基本相同。

3. 多核 CPU

核心(Die)又称为内核，是 CPU 最重要的组成部分。多核即在一个单芯片上面集成两个

甚至更多个处理器内核，其中每个内核都有自己的逻辑单元、控制单元、中断处理器、运算单元，一级 Cache(高速缓存)、二级 Cache 共享或独有，其部件的完整性和单核处理器内核相比完全一致。

1.1.1.3 数据表示

各种数据在计算机中表示的形式称为机器数，其特点是数的符号用 0、1 表示。机器数对应的实际数值称为该数的真值。机器数又分为无符号数和带符号数两种。无符号数表示正数，在机器数中没有符号位。对于带符号数，机器数的最高位是表示正、负的符号位，其余二进制位表示数值。带符号的机器数可采用原码、反码、补码、移码等编码方法。机器数的这些编码方法称为码制。

1. 原码、反码、补码和移码

1) 原码

在原码表示中，机器数的最高位是符号位，0 代表正号，1 代表负号，余下各位是数的绝对值。零有两个编码，即$[+0]_{原}=00000000$、$[-0]_{原}=10000000$。原码表示方法的优点在于数的真值和它的原码表示之间的对应关系简单，相互转换容易，用原码实现乘、除运算的规则简单。缺点是用原码实现加、减运算很不方便。

2) 反码

在反码表示中，机器数的最高一位是符号位，0 代表正号，1 代表负号。当符号位为 0 时，其余几位即为此数的二进制值；但若符号位为 1 时，则要把其余几位按位取反，才是它的二进制值。零有两个编码，即$[+0]_{反}=00000000$、$[-0]_{反}=11111111$。

3) 补码

在补码表示中，机器数的最高一位是符号位，0 代表正号，1 代表负号。当符号位为 0(即正数)时，其余几位即为此数的二进制值；但若符号位为 1(即负数)时，其余几位不是此数的二进制值，需把它们按位取反，且最低位加 1，才是它的二进制值。零有唯一的编码，即$[+0]_{补}=[-0]_{补}=00000000$。补码表示的两个数在进行加法运算时，只要结果不超出机器所能表示的数值范围，可以把符号位与数值位同等处理，运算后的结果按 2 取模后，得到的新结果就是本次加法运算的结果。

4) 移码

移码表示法是在数 X 上增加一个偏移量来定义的，常用于表示浮点数中的阶码。如果机器字长为 n，规定偏移量为 2^{n-1}，则移码定义为：若 X 是纯整数，则$[X]_{移}=2^{n-1}+X(-2^{n-1}\leqslant X<2^{n-1})$；若 X 是纯小数，则$[X]_{移}=1+X(-1\leqslant X<1)$。

2. 定点数和浮点数

1) 定点数

定点数是指小数点的位置固定不变的数。小数点的位置通常有两种约定方式：定点整数(纯整数，小数点在最低有效数值位之后)和定点小数(纯小数，小数点在最高有效数值位之前)。

2) 浮点数

浮点数是小数点位置不固定的数，它能表示更大范围的数。在浮点表示法中，阶码通常为带符号的纯整数，尾数为带符号的纯小数。浮点数的表示格式如图 1-1 所示。

阶符	阶码	数符	尾数

图 1-1　浮点数的表示格式

浮点数通常表示成

$$N = MR^E$$

式中：M 为尾数；R 为基数；E 为阶码。因此，若表示一个浮点数，要给出尾数 M，它决定了浮点数的表示精度；同时要给出阶码 E，它指出了小数点在数据中的位置，决定了浮点数的表示范围(若表示范围超出了计算机的表达范围时，就称为溢出)。

3)　工业标准 IEEE 754

IEEE 754 是由 IEEE 制定的有关浮点数的工业标准，被广泛采用。该标准的表示形式为：

$$P = (-1)^S 2^E (b_0 b_1 b_2 b_3 \cdots b_{p-1})$$

式中：$(-1)^S$ 为该浮点数的数符，当 S 为 0 时表示正数，当 S 为 1 时表示负数；E 为指数(阶码)，用移码表示；$(b_0 b_1 b_2 b_3 \cdots b_{p-1})$ 为尾数，其长度为 p 位，用原码表示。

1.1.1.4　校验码

计算机系统运行时，各个部件之间要进行数据交换，有两种方法可以确保数据在传送过程中正确无误，一是提高硬件电路的可靠性，二是提高代码的校验能力，包括查错和纠错。通常使用校验码的方法来检测传送的数据是否出错。码距是指一个编码系统中任意两个合法编码之间至少有多少个二进制位不同。

1.　奇偶校验

奇偶校验是一种简单而有效的校验方法。其基本思想是，通过在编码中增加一位校验位来使编码中 1 的个数为奇数(奇校验)或者为偶数(偶校验)，从而使码距变为 2。对于奇校验，它可以检测代码中奇数位出错的编码，但不能发现偶数位出错的情况，即当合法编码中奇数位发生了错误，也就是编码中的 1 变成 0 或 0 变成 1，则该编码中 1 的个数的奇偶性就发生了变化，从而可以发现错误。

常用的奇偶校验码有 3 种，即水平奇偶校验码、垂直奇偶校验码和水平垂直校验码。

2.　海明码

海明码的构成方法是：在数据位之间插入 k 个校验码，通过扩大码距来实现检错和纠错。设数据位是 n 位，校验位是 k 位，则 n 和 k 必须满足 $2^k-1 \geqslant n+k$ 的关系。

3.　循环冗余校验码

循环冗余校验码(CRC)广泛应用于数据通信领域和磁介质存储系统中。它利用生成多项式为 k 个数据位产生 r 个校验位来进行编码，其编码长度为 $k+r$。CRC 的代码格式如图 1-2 所示。

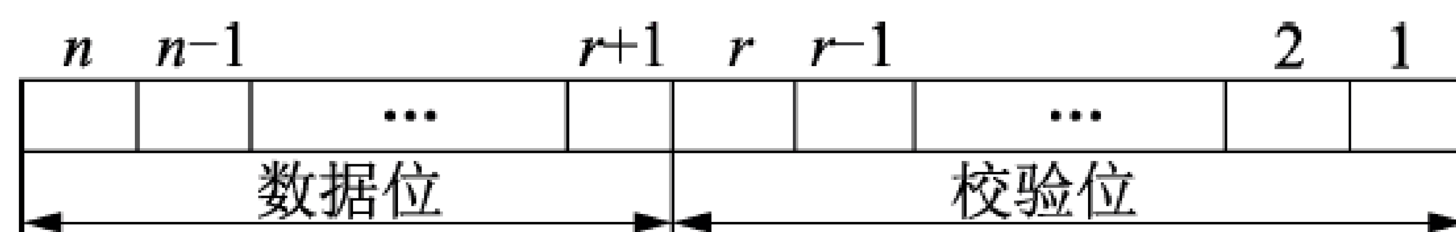


图 1-2　CRC 的代码格式

由此可知，循环冗余校验码是由两部分组成的，左边为信息码(数据)，右边为校验码。若信息码占 k 位，则校验码就占 $n-k$ 位。其中，n 为 CRC 码的字长，所以又称为(n, k)码。校验码是由信息码产生的，校验码位数越长，该代码的校验能力就越强。在求 CRC 编码时，采用的是模 2 运算。模 2 运算加减运算的规则是按位运算，不发生借位和进位。

1.1.2 典型例题分析

例 1 常用的虚拟存储器由(1)两级存储器组成。(2013 年上半年试题 1)

(1) A. 主存—辅存　　B. 主存—网盘

C. Cache—主存　　D. Cache—硬盘

解析：虚拟存储器是为了给用户提供更大的随机存取空间而采用的一种存储技术。它将内存与外存结合使用，好像有一个容量极大的内存储器，工作速度接近于主存，每位成本又与辅存相近，在整机形成多层次存储系统。所以虚拟存储器由主存和辅存两级存储器组成。

答案：A

例 2 在 CPU 中，(1)不仅要保证指令的正确执行，还要能够处理异常事件。(2012 年下半年试题 1)

(1) A. 运算器　　B. 控制器　　C. 寄存器组　　D. 内部总线

解析：控制器负责完成协调和指挥整个计算机系统的操作，是发布命令的决策机构。运算器是数据加工部件，负责执行算术运算和逻辑运算。寄存器一般用来保存程序的中间结果，为随后的指令快速提供操作数，从而避免把中间结果存入内存，再读取内存的操作。内部总线将处理器内部的所有结构单元相连。

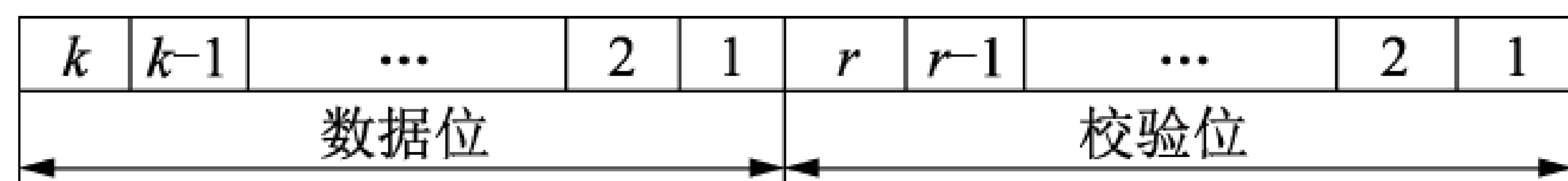
答案：B

例 3 循环冗余校验码(CRC)利用生成多项式进行编码。设数据位为 k 位，校验位为 r 位，则 CRC 码的格式为(2)。(2012 年下半年试题 2)

(2) A. k 个数据位之后跟 r 个校验位　　B. r 个校验位之后跟 k 个数据位

C. r 个校验位随机加入 k 个数据位中　　D. r 个校验位等间隔地加入 k 个数据位中

解析：循环冗余校验码利用生成多项式为 k 个数据位产生 r 个校验位来进行编码，其编码长度为 $k+r$。编码的格式为：



答案：A

例 4 以下关于数的定点表示和浮点表示的叙述中，不正确的是(3)。(2012 年下半年试题 3)

(3) A. 定点表示法表示的数(称为定点数)常分为定点整数和定点小数两种

B. 定点表示法中，小数点需要占用一个存储位

C. 浮点表示法用阶码和尾数来表示数，称为浮点数

D. 在总位数相同的情况下，浮点表示法可以表示更大的数

解析：定点数是小数点位置固定不变的数。定点数分为定点整数和定点小数，前者约

定小数点在最低有效值位之后，后者约定小数点在最高有效值位之前。可见，在定点表示法中，小数点不需要占用存储位。

答案：B

例 5　在程序执行过程中，Cache 与主存的地址映像由__(1)__。(2013 年下半年试题 1)

(1)　A．硬件自动完成
　　B．程序员调度
　　C．操作系统管理
　　D．程序员与操作系统协同完成

解析：CPU 对存储器的访问，通常是一次读写一个字单元。当 CPU 访问 Cache 不命中时，需将存储在主存中的字单元连同其后若干个字一同调入 Cache 中。由于 CPU 首先访问的是 Cache，并不是主存。为此，需要一种机制将 CPU 的访主存地址转换成访 Cache 地址。而主存地址与 Cache 地址之间的转换是和主存块与 Cache 块之间的映射关系紧密联系的。

答案：A

例 6　对于逻辑表达式“x and y or not z”，and、or、not 分别是逻辑与、或、非运算，优先级从高到低为 not、and、or，and、or 为左结合，not 为右结合，若进行短路计算，则__(20)__。(2012 年上半年试题 20)

(20)　A. x 为真时，整个表达式的值为真，不需要计算 y 和 z 的值
　　B. x 为假时，整个表达式的值为假，不需要计算 y 和 z 的值
　　C. x 为真时，根据 y 的值决定是否需要计算 z 的值
　　D. x 为假时，根据 y 的值决定是否需要计算 z 的值

解析：“x and y or not z”可以表示为“(x and y) or (not z)”。x 为真时，如果 y 为真，则 x and y 的结果为真，此时不需要计算 z 的值，整个表达式的值为真；如果 y 为假，则 x and y 的结果为假，此时整个表达式的值由 z 决定，如果 z 为真，则 not z 为假，最终结果为假，否则为真。

x 为假时，不需要计算 y 的值，x and y 的结果为假，整个表达式的值由 z 决定。

答案：C

例 7　若某条无条件转移汇编指令采用直接寻址，则该指令的功能是将指令中的地址码送入__(1)__。(2011 年下半年试题 1)

(1)　A．PC(程序计数器)　　B．AR(地址寄存器)
　　C．ACC(累加器)　　D．ALU(算术逻辑单元)

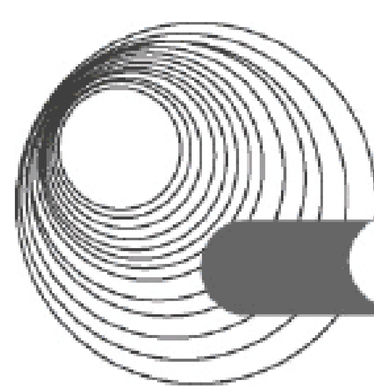
解析：若某条无条件转移汇编指令采用直接寻址，则该指令的功能是将指令中的地址码送入程序计数器。也可以用排除法。直接寻址：操作数的有效地址直接在指令中给出，所以不需要地址寄存器，B 错误；ACC(累加器)是存放运算过程中的中间结果的，C 错误；ALU(算术逻辑运算单元)是运算器组成部分，此题目不涉及。

答案：A

例 8　计算机中 CPU 对其访问速度最快的是__(1)__。(2015 年上半年试题 1)

(1)　A．内存　　B．Cache
　　C．通用寄存器　　D．硬盘

解析：题目中 4 种存储设备按访问速度排序为：通用寄存器>Cache>内存>硬盘。



答案：(1)C

例 9 CPU 中译码器的主要作用是进行(6)。(2011 年下半年试题 6)

(6) A. 地址译码　　B. 指令译码

C. 数据译码　　D. 选择多路数据至 ALU

解析：指令译码器是控制器中的主要部件之一。计算机能且只能执行“指令”。指令由操作码和操作数组成。操作码表示要执行的操作性质，即执行什么操作或做什么；操作数是操作码执行时的操作对象，即对什么数进行操作。计算机执行一条指定的指令时，必须首先分析这条指令的操作码是什么，以决定操作的性质和方法，然后才能控制计算机其他各部件协同完成指令表达的功能。这个分析工作由译码器来完成。

答案：B

例 10 在 CPU 中用于跟踪指令地址的寄存器是(1)。(2011 年上半年试题 1)

(1) A. 地址寄存器(MAR)　　B. 数据寄存器(MDR)

C. 程序计数器(PC)　　D. 指令寄存器(IR)

解析：程序计数器是用于存放下一条指令所在单元的地址的地方，执行指令时，CPU 将自动修改程序计数器的内容，因此，在 CPU 中用于跟踪指令地址的寄存器就是程序计数器。地址寄存器保存当前 CPU 所访问的内存单元的地址；当 CPU 要执行一条指令时，先把它从内存取到数据缓冲寄存器中，再送入指令寄存器 IR 暂存，指令译码器根据 IR 的内容产生各种微操作命令，控制其他的组成部件工作，完成所需的功能。

答案：C

例 11 原码表示法和补码表示法是计算机中用于表示数据的两种编码方法，在计算机系统中常采用补码来表示和运算数据，原因是采用补码可以(4)。(2011 上半年试题 4)

(4) A. 保证运算过程与手工运算方法保持一致

B. 简化计算机运算部件的设计

C. 提高数据的运算速度

D. 提高数据的运算精度

解析：使用补码，可以将符号位和其他位统一处理，同时，减法也可以按照加法来处理。另外，两个用补码表示的数相加时，如果最高位有进位，则进位会被舍弃。可见，采用补码可以简化运算及其电路。

答案：B

例 12 机器字长为 n 位的二进制数可以用补码来表示 (2) 个不同的有符号定点小数。(2015 年上半年试题 2)

(2) A. $2n$　　B. $2n-1$　　C. $2n-1$　　D. $2n-1+1$

解析：机器字长为 n 时，补码可表示的定点小数范围为 $-1\sim+\left(1-2^{-(n-1)}\right)$，共有 2^n 个数。

答案：A

例 13 某机器字长为 n，最高位是符号位，其定点整数的最大值为 (2) 。(2014 年上半年试题 2)

(2) A. 2^n-1　　B. $2^{n-1}-1$　　C. 2^n　　D. 2^{n-1}

解析：由于最高位是符号位，因此最大的定点整数是：

$$\underbrace{011111\cdots\cdots111}_{n-1\text{ 个 }1}$$

最高位 0 表示正数，值为 $2^0+2^1+2^2+\cdots+2^{n-2}=2^{n-1}-1$。

答案：B

例 14　属于 CPU 中算术逻辑单元的部件是__(3)__。(2014 年下半年试题 3)

(3)　A. 程序计数器　　B. 加法器
　　C. 指令寄存器　　D. 指令译码器

解析：算术逻辑运算单元 ALU(Arithmetic and Logic Unit)主要完成对二进制数据的定点算术运算(加、减、乘、除)、逻辑运算(与、或、非、异或)以及移位操作。

答案：B

例 15　与 $\overline{A}\oplus B$ 等价的逻辑表达式是__(4)__。(⊕表示逻辑异或，+表示逻辑加)。(2010 年上半年试题 4)

(4) A. $A+\overline{B}$　　B. $A\oplus\overline{B}$　　C. $A\oplus B$　　D. $AB+\overline{AB}$

解析：用真值表验证如下：

A	B	选项 A $A+\overline{B}$	选项 B $A\oplus\overline{B}$	选项 C $A\oplus B$	选项 D $AB+\overline{AB}$	$\overline{A}\oplus B$
0	0	1	1	0	1	1
0	1	0	0	1	1	0
1	0	1	0	1	1	0
1	1	1	1	0	1	1

从上表可知，$\overline{A}\oplus B$ 与 $A\oplus\overline{B}$ 等价。

答案：B

例 16　关于 64 位和 32 位微处理器，不能以 2 倍关系描述的是__(6)__。(2010 年上半年试题 6)

(6) A. 通用寄存器的位数　　B. 数据总线的宽度
　　C. 运算速度　　D. 能同时进行运算的位数

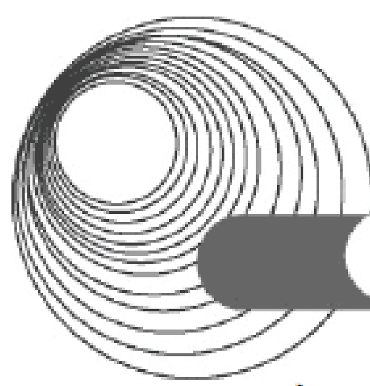
解析：计算机系统的运算速度受多种因素的影响，64 位微处理器可同时对 64 位数据进行运算，但不能说其速度是 32 位微处理器的 2 倍。

答案：C

例 17　若某整数的 16 位补码为 FFFFH(H 表示十六进制)，则该数的十进制值为__(20)__。(2010 年上半年试题 20)

(20) A. 0　　B. −1　　C. 216−1　　D. −216+1

解析：根据补码定义，数值 X 的补码记作 $[X]_{补}$，如果机器字长为 n，则最高位为符号位，0 表示正号，1 表示负号，正数的补码与其原码和反码相同，负数的补码则等于其反码的末尾加 1。如果已知 X 的补码为 FFFFH，对应的二进制数为 1111111111111111，则 X 的反码



为 1111111111111110，*X* 的原码为 1000000000000001，对应的十进制数为-1。

答案：B

例 18　以下关于 CPU 的叙述中，错误的是<u> (1) </u>。(2009 年下半年试题 1)

(1) A．CPU 产生每条指令的操作信号并将操作信号送往相应的部件进行控制

B．程序计数器(PC)除了存放指令地址外，也可以临时存储算术/逻辑运算结果

C．CPU 中的控制器决定计算机运行过程的自动化

D．指令译码器是 CPU 控制器中的部件

解析：本题主要考查 CPU 的组成及其部件的功能。

CPU 的功能主要包括程序控制、操作控制、时间控制和数据处理。CPU 主要由运算器、控制器、寄存器组和内部总线等部件组成。CPU 产生每条指令的操作信号并将操作信号送往相应的部件进行控制，因此说法 A 正确。

CPU 中的控制器用于控制整个 CPU 的工作，它决定了计算机运行过程中的自动化，因此说法 C 正确。

程序计数器(PC)具有寄存信息和计数两种功能，又称为指令计数器。程序的执行分为两种情况，即顺序执行和转移执行。在程序执行前，将程序的起始地址送入 PC，该地址在程序加载到内存时确定，执行指令时，CPU 将自动修改 PC 的内容，当指令按照顺序执行时，PC 加 1。如果是转移指令，后继指令的地址可根据当前指令的地址加上一个向前或向后转移的位移量得到。因此，PC 没有临时存储算术/逻辑运算结果的功能。因此说法 B 错误。

CPU 中的控制器包括指令寄存器(IR)、程序计数器(PC)、地址寄存器(AR)和指令译码器(ID)。因此说法 D 正确。

答案：B

例 19　浮点数能够表示的数的范围是由其<u> (3) </u>的位数决定的。(2015 年下半年试题 3)

(3) A．尾数　　B．阶　　C．数符　　D．阶符

解析：浮点数表示为 $N = 2^E F$，E 为阶码，决定浮点数所能表示的数值范围；F 为尾数，决定浮点数所能表示的数值精度。

答案：(3)B

例 20　以下关于校验码的叙述中，正确的是<u> (5) </u>。(2009 年下半年试题 5)

(5) A．海明码利用多组数位的奇偶性来检错和纠错

B．海明码的码距必须大于等于 1

C．循环冗余校验码具有很强的检错和纠错能力

D．循环冗余校验码的码距必定为 1

解析：本题主要考查海明码和循环冗余校验码。

海明码是由贝尔实验室的 Richard Hamming(理查德·海明)设计的，它也是利用奇偶性来检错和纠错的校验方法。其构成方法是：在数据位之间插入 *k* 个校验位，通过扩大码距来实现检错和纠错。

循环冗余校验(Cyclic Redundancy Check，CRC)码广泛用在数据通信领域和磁介质存储系统中，它利用生成多项式为 *k* 个数据位产生 *r* 个校验位来进行编码，其编码长度为 *k*+*r*。其由两部分组成，左边为信息码(数据)，右边为校验码。若信息码占 *k* 位，则校验码占 *n*−*k* 位。其中，*n* 为 CRC 码的字长，所以又称为(*n*,*k*)码。校验码由信息码产生，校验码位数越长，该代码的校验能力就越强。

答案：A

例 21　海明码利用奇偶性检错和纠错，通过在 n 个数据位之间插入 k 个检验位，扩大数据编码的码距。若 n=48，则 k 应为<u> (3) </u>。(2014 年上半年试题 3)

(3) A. 4　　B. 5　　C. 6　　D. 7

解析：n 和 k 满足关系：

$$2^k-1\geqslant n+k$$

题中 n=48，k 为 6 可满足上述关系式。

答案：C

例 22　计算机中常采用原码、反码、补码和移码表示数据，其中，±0 编码相同的是<u>(5)</u>。(2009 年上半年试题 5)

(5) A．原码和补码　　B．反码和补码
C．补码和移码　　D．原码和移码

解析：原码、反码、补码以及移码是计算机的数据表示形式，需掌握牢固，+0 和−0 的表示比较特殊，在此做个总结。

原码：$[+0]_{原}$=0 0000000　$[-0]_{原}$=1 0000000

反码：$[+0]_{反}$=0 0000000　$[-0]_{反}$=1 1111111

补码：$[+0]_{补}=[-0]_{补}$=0 0000000

移码：$[+0]_{移}=[-0]_{移}$=1 0000000

答案：C

1.1.3　同步练习

1.　计算机在进行浮点数的相加(减)运算之前先进行对阶操作，若 x 的阶码大于 y 的阶码，则应将______。

A．x 的阶码缩小至与 y 的阶码相同，且使 x 的尾数部分进行算术左移
B．x 的阶码缩小至与 y 的阶码相同，且使 x 的尾数部分进行算术右移
C．y 的阶码扩大至与 x 的阶码相同，且使 y 的尾数部分进行算术左移
D．y 的阶码扩大至与 x 的阶码相同，且使 y 的尾数部分进行算术右移

2.　在 CPU 中，_____可用于传送和暂存用户数据，为 ALU 执行算术逻辑运算提供工作区。

A．程序计数器　B．累加寄存器　C．程序状态寄存器　D．地址寄存器

3.　下面关于校验方法的叙述，正确的是______。

A．采用奇偶校验可检测数据传输过程中出现一位数据错误的位置并加以纠正
B．采用海明校验可检测数据传输过程中出现一位数据错误的位置并加以纠正
C．采用海明校验，校验码的长度和位置可随机设定
D．采用 CRC 校验，需要将校验码分散开并插入数据的指定位置中

4.　在计算机体系结构中，CPU 内部包括程序计数器(PC)、存储器数据寄存器(MDR)、指令寄存器(IR)和存储器地址寄存器(MAR)等。若 CPU 要执行的指令为 MOV R0，# 100(即将数值 100 传送到寄存器 R0 中)，则 CPU 首先要完成的操作是______。

A．100→R0　　B．100→MDR　　C．PC→MAR　　D．PC→IR

5. 指令寄存器的位数取决于______。(2013 年下半年试题 2)

A．存储器的容量　　　B．指令字长

C．数据总线的宽度　　D．地址总线的宽度

1.1.4 同步练习参考答案

1．D　　2．B　　3．B　　4．C　　5．B

1.2 计算机体系结构

1.2.1 考点辅导

1.2.1.1 计算机体系结构的发展

关于计算机体系结构、计算机组织和计算机实现三者的关系如下。

(1) 计算机体系结构(Computer Architecture)是指计算机的概念性结构和功能属性。

(2) 计算机组织(Computer Organization)是指计算机体系结构的逻辑实现，包括机器内的数据流和控制流的组成以及逻辑设计等(常称为计算机组成原理)。

(3) 计算机实现(Computer Implementation)是指计算机组织的物理实现。

1. 计算机体系结构的分类

1) 宏观上的分类(按处理机的数量分类)

(1) 单处理系统(Uniprocessing System)，是指利用一个处理单元与其他外部设备结合起来，实现存储、计算、通信、输入与输出等功能的系统。

(2) 并行处理与多处理系统(Parallel Processing and Multiprocessing System)，是指为了充分发挥问题求解过程中处理的并行性，将两个以上的处理机相互连接起来，彼此进行通信协调，以便共同求解一个大问题的计算机系统。

(3) 分布式处理系统(Distributed Processing System)，是指物理上远距离而松耦合的多计算机系统。其中，物理上的远距离意味着通信时间与处理时间相比已不可忽略，在通信线路上的数据传输速率要比在处理机内部总线上的传输慢得多，这也正是松耦合的含义。

2) 微观上的分类(按并行程度分类)

(1) Flynn 分类法。

1966 年，M. J. Flynn 提出按指令流和数据流的不同组织方式把计算机体系结构分为以下四大类。

- 单指令流单数据流(SISD)。
- 单指令流多数据流(SIMD)。
- 多指令流单数据流(MISD)。
- 多指令流多数据流(MIMD)。

(2)　冯译云分类法。

1972 年，美籍华人科学家冯译云提出按最大并行度来进行分类。最大并行度是指计算机系统在单位时间内能够处理的最大的二进制位数。冯译云将计算机系统结构分为以下 4 种。

- 字串行位串行(WSBS)。
- 字并行位串行(WPBS)。
- 字串行位并行(WSBP)。
- 字并行位并行(WPBP)。

(3)　Handler 分类法。

1977 年，德国的汉德勒(Wolfgang Handler)提出一个基于硬件并行程序计算并行度的方法，把计算机的硬件结构分为以下 3 个层次。

- 处理机级。
- 每个处理机中的算术逻辑单元级。
- 每个算术逻辑单元中的逻辑门电路级。

(4)　Kuck 分类法。

1978 年，美国的库克(David J. Kuck)提出与 Flynn 分类法类似的方法，用指令流和执行流及其多重性来描述计算机系统控制结构的特征。Kuck 把系统结构分为以下 4 类。

- 单指令流单执行流(SISE)。
- 单指令流多执行流(SIME)。
- 多指令流单执行流(MISE)。
- 多指令流多执行流(MIME)。

2.　指令系统

指令系统指的是一个 CPU 所能够处理的全部指令的集合，是一个 CPU 的根本属性。一条指令一般包括两个部分，即操作码和地址码。操作码指明操作的类型，地址码主要指明操作数及运算结果存放的地址。

1)　寻址方式

表示指令中操作数所在的方法称为寻址方式。常见的寻址方式有以下几种。

(1)　立即寻址：操作数作为指令的一部分而直接写在指令中，这种操作数称为立即数。

(2)　寄存器寻址：指令所要的操作数已存储在某寄存器中，或把目标操作数存入寄存器。

(3)　直接寻址：指令所要的操作数存放在内存中，在指令中直接给出该操作数的有效地址。

(4)　寄存器间接寻址：操作数在存储器中，操作数的有效地址用 SI、DI、BX 和 BP 这 4 个寄存器之一来指定。

(5)　寄存器相对寻址：操作数在存储器中，其有效地址是一个基址寄存器(BX、BP)或变址寄存器(SI、DI)的内容和指令中的 8 位/16 位偏移量之和。

(6)　基址加变址寻址方式：操作数在存储器中，其有效地址是一个基址寄存器(BX、BP)和一个变址寄存器(SI、DI)的内容之和。

(7)　相对基址加变址寻址：操作数在存储器中，其有效地址是一个基址寄存器(BX、BP)的值、一个变址寄存器(SI、DI)的值和指令中的 8 位/16 位偏移量之和。

2) CISC和RISC

(1) CISC(Complex Instruction Set Computer，复杂指令集计算机)的基本思想是：进一步增强原有指令的功能，用更为复杂的新指令取代原先由软件子程序完成的功能，实现软件功能的硬化，导致机器的指令系统越来越庞大而复杂。

CISC的主要弊病如下。

① 指令集过分庞杂。

② 微程序技术是CISC的重要支柱，每条复杂指令都要通过执行一段解释性微程序才能完成，这就需要多个CPU周期，从而降低了机器的处理速度。

③ 由于指令系统过分庞大，使高级语言编译程序选择目标指令的范围很大，并使编译程序本身冗长而复杂，从而难以优化编译使之生成真正高效的目标代码。

④ CISC强调完善的中断控制，势必导致动作繁多、设计复杂、研制周期长。

⑤ CISC给芯片设计带来了很多困难，使芯片种类增多、出错概率增大、成本提高而成品率降低。

(2) RISC(Reduced Instruction Set Computer，精简指令集计算机)的基本思想是：通过减少指令总数和简化指令功能，降低硬件设计的复杂度，使指令能单周期执行，并通过优化编译，提高指令的执行速度，采用硬线控制逻辑，优化编译程序。

RISC的关键技术如下。

① 重叠寄存器窗口技术。在加州大学伯克利分校的RISC项目中，首先采用了重叠寄存器窗口(Overlapping Register Windows)技术。

② 优化编译技术。RISC使用了大量的寄存器，如何合理分配寄存器、提高寄存器的使用效率及减少访存次数等，都应通过编译技术的优化来实现。

③ 超流水及超标量技术。这是RISC为了进一步提高流水线速度而采用的技术。

④ 硬布线逻辑与微程序在微程序技术中相结合。

3. 指令的流水处理

1) 指令控制方式

指令控制方式有顺序方式、重叠方式和流水方式3种。

(1) 顺序方式。顺序方式是指各条机器指令之间顺序串行地执行，执行完一条指令后才取下一条指令，而且每条机器指令内部的各个微操作也是顺序串行地执行。这种方式的优点是控制简单；缺点是速度上不去，机器各部件的利用率低。

(2) 重叠方式。重叠方式是指在解释第 x 条指令的操作完成之前，就可开始解释第 $x+i$ 条指令。通常采用的是一次重叠，即在任何时候，指令分析部件和指令执行部件都只有相邻两条指令在重叠解释。这种方式的优点是速度有所提高，控制也不太复杂；缺点是会出现冲突、转移和相关等问题，在设计时必须想办法解决。

(3) 流水方式。流水技术是把并行性或并发性嵌入到计算机系统里的一种形式，它把重复的顺序处理过程分解为若干子过程，每个子过程能在专用的独立模块上有效地并发工作，如图1-3所示。

在概念上，“流水”可以看成是“重叠”的延伸。差别仅在于“一次重叠”只是把一条指令解释分解为两个子过程，而“流水”则是分解为更多的子过程。

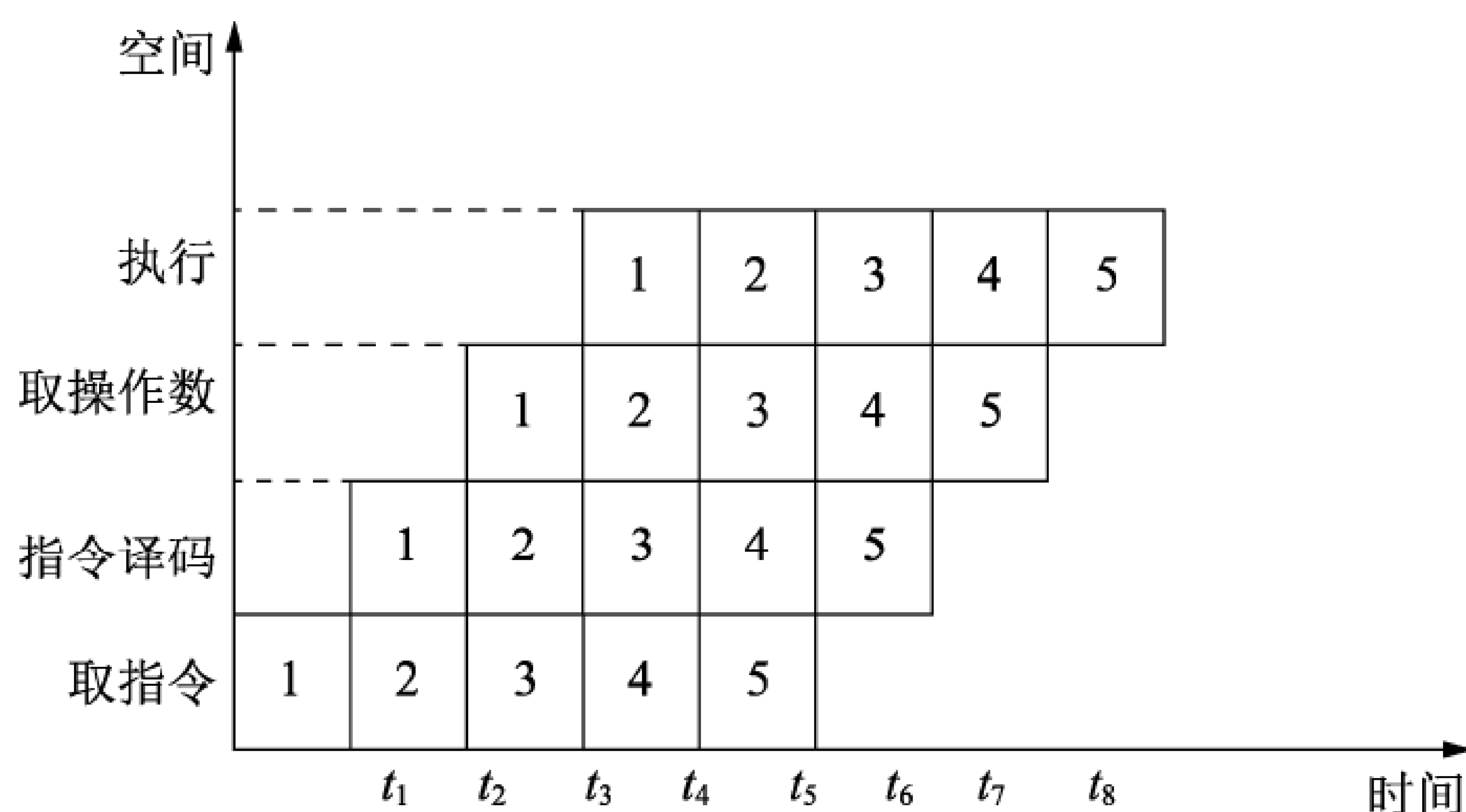


图 1-3　流水处理的时空图

2)　吞吐率和流水建立时间

吞吐率是指单位时间里流水线处理机流出的结果数。对指令而言，就是单位时间里执行的指令数。如果流水线的子过程所用时间不一样，则吞吐率 p 应为最长子过程的倒数，即

$$p=\frac{1}{\max\{\Delta t_1,\Delta t_2,\cdots,\Delta t_m\}}$$

流水线开始工作，须经过一定时间才能达到最大吞吐率，这就是建立时间。若 m 个子过程所用时间一样，均为Δt_0，则建立时间 $T_0=m\Delta t_0$。

4. 阵列处理机、并行处理机和多处理机

并行性包括同时性和并发性两个侧面。其中，同时性是指两个或两个以上的事件在同一时刻发生，并发性是指两个或两个以上的事件在同一时间间隔内连续发生。

从计算机信息处理的步骤和阶段的角度看，并行处理可分为以下几类。

- 存储器操作并行。
- 处理器操作步骤并行(流水线处理机)。
- 处理器操作并行(阵列处理机)。
- 指令、任务、作业并行(多处理机、分布式处理系统、计算机网络)。

1)　阵列处理机

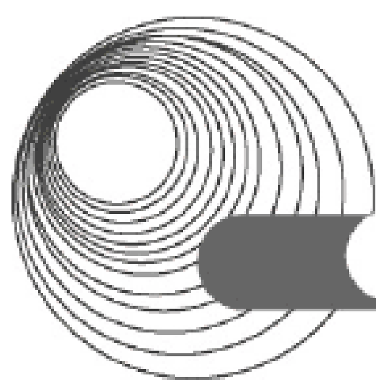
阵列处理机将重复设置的多个处理单元(PU)按一定方式连成阵列，在单个控制部件(CU)控制下，对分配给自己的数据进行处理，并行地完成一条指令所规定的操作。这是一种单指令流多数据流计算机，通过资源重复实现并行性。

2)　并行处理机

SIMD 和 MIMD 是典型的并行计算机，SIMD 有共享存储器和分布存储器两种形式。

具有共享存储器的 SIMD 结构(见图 1-4)中，将若干个存储器构成统一的并行处理机存储器，通过互连网络(Interconnection Network，ICN)为整个并行系统的所有处理单元共享。其中，PE 为处理单元，CU 为控制部件，M 为共享存储器，ICN 为互连网络。

分布存储器的 SIMD 处理机如图 1-5 所示，其中 PE 为处理单元，CU 为控制部件，PEM 为局部存储器，ICN 为互连网络。含有多个同样结构的处理单元，通过寻径网络 ICN 以一定方式互相连接。



分布存储器的并行处理机结构中有两类存储器：一类存储器附属于主处理机，主处理机实现整个并行处理机的管理，在其附属的存储器内常驻操作系统；另一类是分布在各个处理单元上的存储器(即 PEM)，这类存储器用来保存程序和数据。

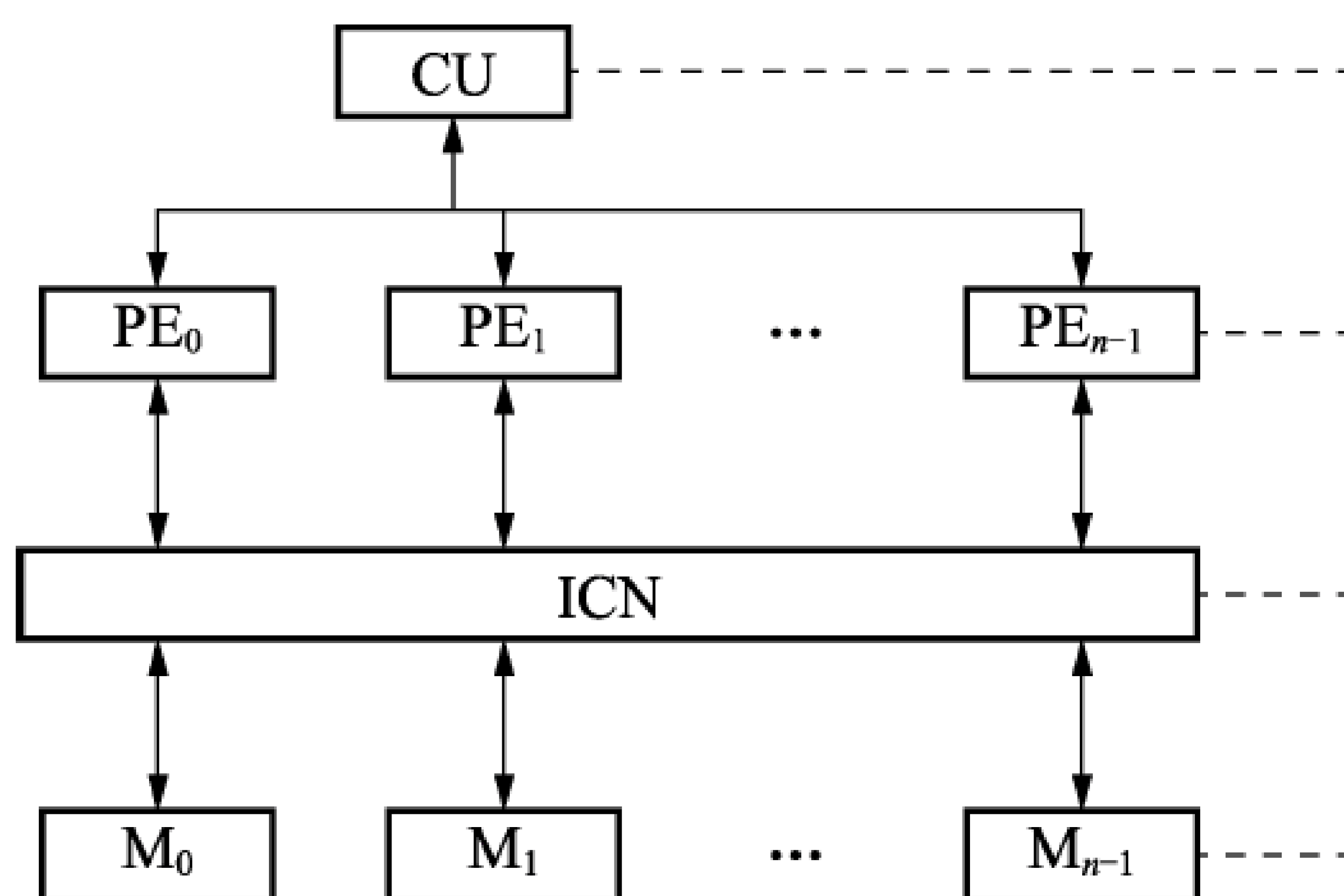


图 1-4　具有共享存储器的 SIMD 结构

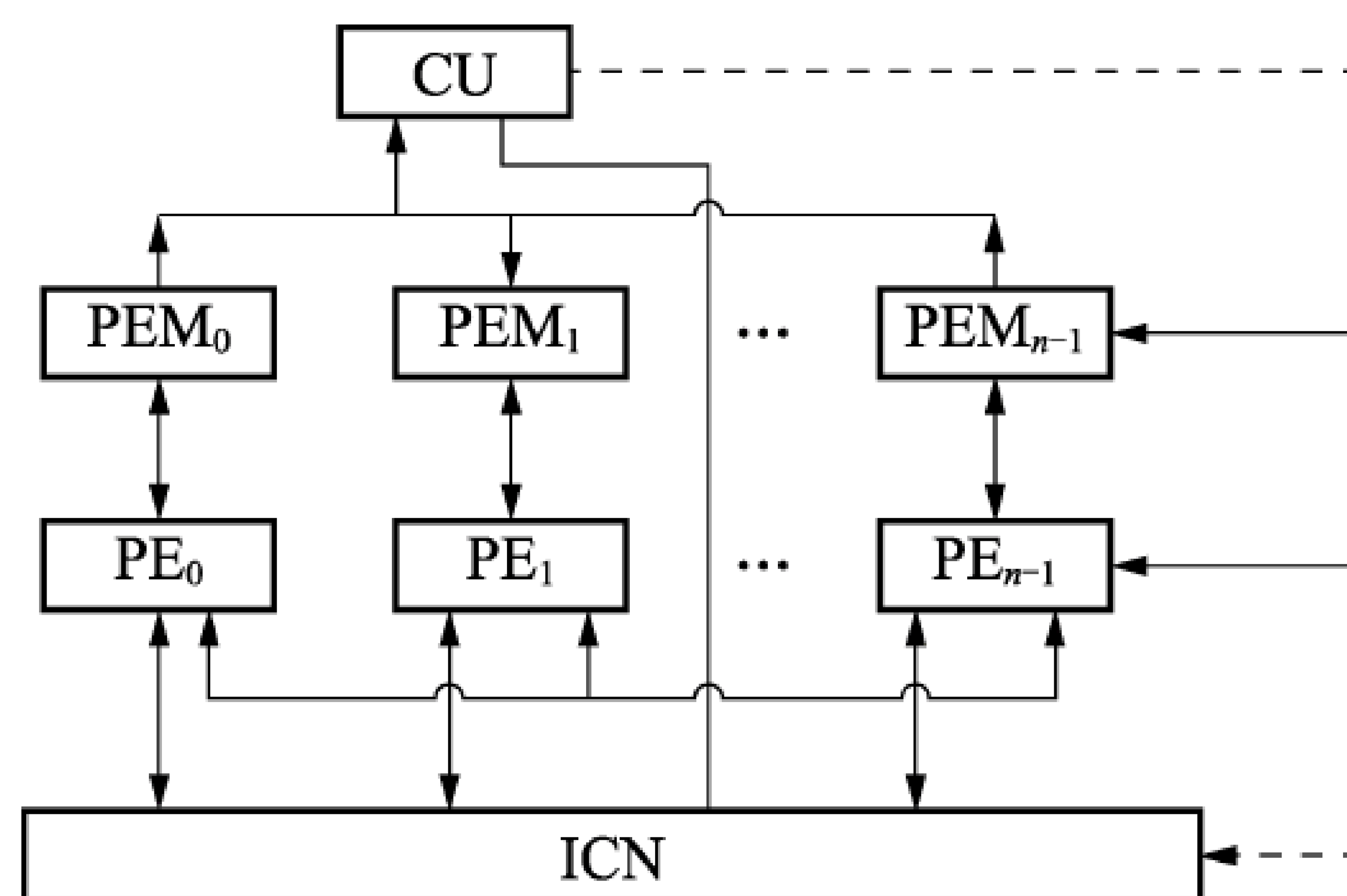


图 1-5　具有分布存储器的 SIMD 结构

3)　多处理机

多处理机系统是由多台处理机组成的系统，每台处理机都有属于自己的控制部件，可以执行独立的程序，共享一个主存储器和所有的外部设备。它是 MIMD 计算机。多处理机之间的互连，要满足高带宽、低成本、连接方式的多样性以及在不规则通信情况下连接的无冲突性。

1.2.1.2　存储系统

1.　*存储器的层次结构*

计算机的三层存储体系结构如图 1-6 所示。

三层存储结构是高速缓存(Cache)、主存储器(Main Memory，MM)和辅助存储器(外存储器)。若将 CPU 内部寄存器也看作存储器的一个层次，那么存储器的层次分为 4 层。若有些计算机没有高速缓存，那么存储器的层次分为两层，即只有主存和辅存。

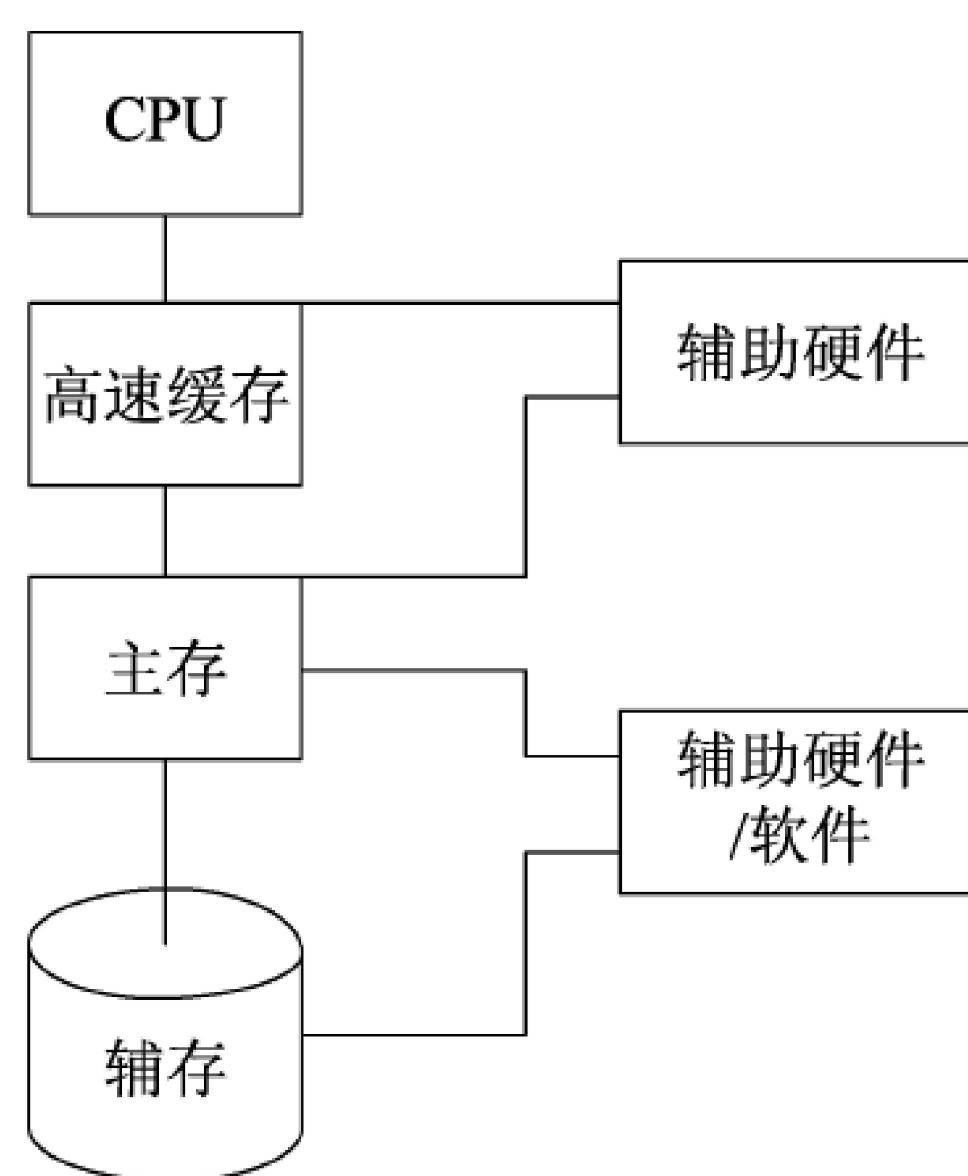


图 1-6　存储器层次结构示意框图

2.　*存储器的分类*

1)　按位置分类

存储器按位置分类，可分为内存和外存。

(1)　内存(主存)：用来存储当前运行所需要的程序和数据，速度快，容量小。

(2)　外存(辅存)：用来存储目前不参与运行的数据，容量大但速度慢。

2)　按材料分类

存储器按材料分类，可分为磁存储器、半导体存储器和光存储器。

(1)　磁存储器：用磁性介质做成的，如磁芯、磁泡、磁盘、磁带等。

(2)　半导体存储器：根据所用元件又可分为双极型和 MOS 型；根据是否需要刷新又可分为静态和动态两类。

(3)　光存储器：由光学、电学和机械部件等组成，如光盘存储器。

3)　按工作方式分类

存储器按工作方式分类，可分为读写存储器和只读存储器。

(1)　读写存储器：既能读取数据也能存入数据的存储器。

(2)　只读存储器：根据数据写入方式，又可细分为固定只读存储器、可编程只读存储器、可擦除可编程只读存储器、电擦除可编程只读存储器和闪速存储器。

4)　按访问方式分类

存储器按访问方式分类，可分为按地址访问的存储器和按内容访问的存储器。

5)　按寻址方式分类

存储器按寻址方式分类，可分为随机存储器、顺序存储器和直接存取存储器。

(1)　随机存储器(Random Access Memory，RAM)：这种存储器可对任何存储单元存入或读取数据，访问任何一个存储单元所需时间都是相同的。

(2)　顺序存储器(Sequentially Addressed Memory，SAM)：访问数据所需时间与数据所在存储位置有关，磁带是典型的顺序存储器。

(3)　直接存取存储器(Direct Addressed Memory，DAM)：采用介于随机存取和顺序存取之间的一种寻址方式。磁盘是一种直接存取控制器，它对磁道的寻址是随机的，而在一个磁道内，则是顺序寻址。

3.　相联存储器

相联存储器是一种按内容访问的存储器。其工作原理是把数据或数据的某一部分作为关键字，将该关键字与存储器中的每一单元进行比较，找出存储器中所有与关键字相同的数据字。

高速缓冲存储器(可简称为高速缓存或缓存)可用在相联存储器中，在虚拟存储器中用来作段表、页表或块表存储器，还可以用在数据库和知识库中。

4.　高速缓存

高速缓存(Cache)是位于 CPU 和主存之间的高速存储子系统。采用高速缓存的主要目的是提高存储器的平均访问速度，使存储器的速度与 CPU 的速度相匹配。Cache 的存在对程序员是透明的。其地址变换和数据块的替换算法均由硬件实现。通常 Cache 被集成到 CPU 内，以提高访问速度，其主要特点是容量小、速度快、成本高。

1)　Cache 的组成

Cache 的组成如图 1-7 所示。Cache 由两部分组成，即控制部分和缓存部分。缓存部分用来存放主存的部分复制信息。控制部分的功能是：判断 CPU 要访问的信息是否在 Cache 中，若在即为命中，若不在则没有命中。命中时直接对 Cache 寻址；未命中时，要按照替换原则，决定主存的一块信息放到 Cache 的哪一块里面。

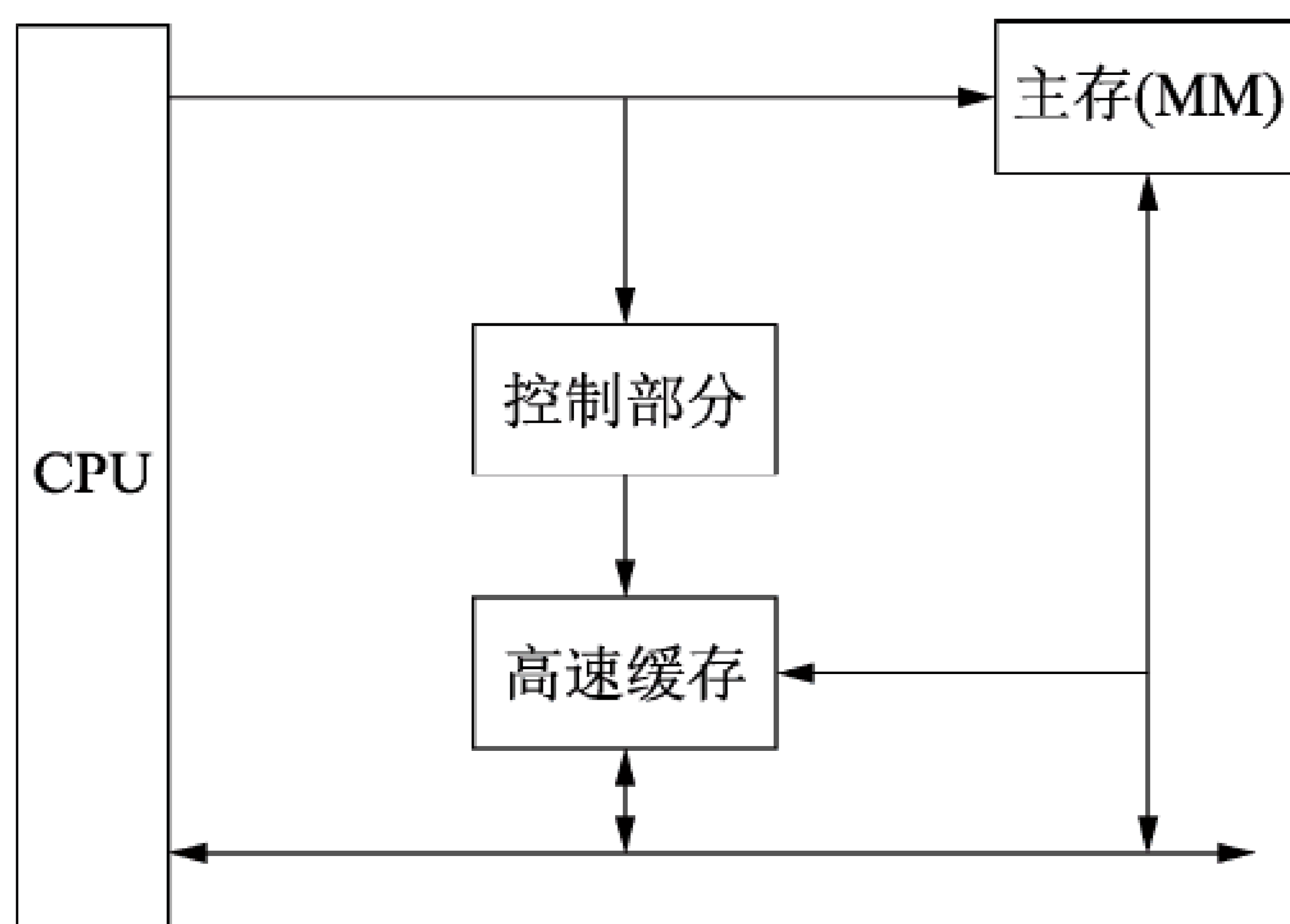
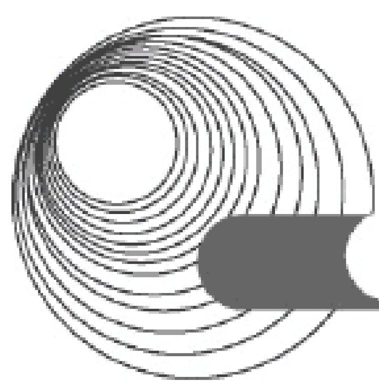


图 1-7　高速缓存的组成框图



2) Cache 中的地址映像方法

因为处理机访问都是按主存地址访问的，而应从 Cache 中读写信息，因此这就需要地址映像，即把主存中的地址映射成 Cache 中的地址。地址映像的方法有 3 种，即直接映像、全相联映像和组相联映像。

(1) 直接映像就是主存的块与 Cache 中块的对应关系是固定的。主存中的块只能存放在 Cache 的相同块号中。因此，只要主存地址中的主存区号与 Cache 中的主存区号相同，则表明访问 Cache 命中。一旦命中，以主存地址中的区内块号立即可得到要访问的 Cache 中的块。这种方式的优点是地址变换很简单，缺点是灵活性差。

(2) 全相联映像允许主存的任一块可以调入 Cache 的任何一块的空间中。在地址变换时，利用主存地址高位表示的主存块号与 Cache 中的主存块号进行比较，若相同则为命中。这种方式的优点是主存的块调入 Cache 的位置不受限制，十分灵活；其缺点是无法从主存块号中直接获得 Cache 的块号，变换比较复杂，速度比较慢。

(3) 组相联映像是前面两种方式的折中。具体做法是将 Cache 中的块再分成组。组相联映像就是规定组采用直接映像方式而块采用全相联映像方式。这种方式下，通过直接映像方式来决定组号，在一组内再用全映像方式来决定 Cache 中的块号。由主存地址高位决定主存区号，与 Cache 中区号比较可决定是否命中。主存后面的地址即为组号，但组块号要根据全相联映像方式，由记录可以决定组内块号。

3) 替换算法

选择替换算法的目标是使 Cache 获得最高的命中率。常用的替换算法有以下几种。

(1) 随机替换(RAND)算法：用随机数发生器产生一个要替换的块号，将该块替换出去。

(2) 先进先出(FIFO)算法：将最先进入的 Cache 信息块替换出去。

(3) 近期最少使用(LRU)算法：将近期最少使用的 Cache 中的信息块替换出去。这种算法比先进先出算法要好些，但此法也不能保证过去不常用的将来也不常用。

(4) 优化替换(OPT)算法：先执行一次程序，统计 Cache 的替换情况。有了这样的先验信息，在第二次执行该程序时便可以用最有效的方式来替换，达到最优的目的。

4) Cache 的性能分析

若 H 为 Cache 的命中率，t_c 为 Cache 的存取时间，t_m 为主存的访问时间，则 Cache 的等效访问时间 t_a 为

$$t_a = Ht_c + (1-H)t_m$$

使用 Cache 比不使用 Cache 的 CPU 访问存储器的速度提高的倍数 r 可以用下式求得，即

$$r = \frac{t_m}{t_a}$$

5. 虚拟存储器

虚拟存储器是由主存、辅存、存储管理单元及操作系统中存储管理软件组成的存储系统。程序员使用该存储系统时，可以使用的内存空间远远大于主存的物理空间，但实际上并不存在那么大的主存，故称其为虚拟存储器。虚拟存储器的空间大小取决于计算机的访存能力而不是实际外存的大小，实际存储空间可以小于虚拟地址空间。从程序员的角度看，外存被看作逻辑存储空间，访问的地址是一个逻辑地址(虚地址)，虚拟存储器使存储系统既具有相当于外存的容量又有接近于主存的访问速度。

虚拟存储器的访问也涉及虚地址与实地址的映像、替换算法等，这与 Cache 中的类似。前面讲的地址映像以块为单位，而在虚拟存储器中，地址映像以页为单位。设计虚拟存储系统需考虑的指标是主存空间利用率和主存的命中率。按存储映像算法，可将虚拟存储器的管理方式分为以下 3 种。

(1) 页式虚拟存储器。以页为信息传送单位的虚拟存储器。为实现页式管理，须建立实页与虚页间的关系表，称为页表；在页表及变换软件的控制下，可将程序的虚拟地址变换为内存的实地址。页式虚拟存储器的优点是：页表硬件少，查表速度快；主存零头少。页式虚拟存储器的缺点是：分页无逻辑意义，不利于存储保护。

(2) 段式虚拟存储器。以程序的逻辑结构形成的段作为主存分配依据的一种管理方法。为实现段式管理，须建立段表；在段地址变换机构及软件的控制下，可将程序的虚拟地址变换为主存的实地址。段式虚拟存储器的优点是：段的界线分明；支持程序的模块化设计；易于对程序段的编译、修改和保护；便于多道程序的共享。段式虚拟存储器的主要缺点是：主存利用率不高，查表速度慢。

(3) 段页式虚拟存储器。这是将段式虚拟存储器和页式虚拟存储器结合的一种管理方式。在这种虚拟存储器中，程序按逻辑结构分段，每一段再分成若干大小固定的页。程序的调入调出是按页进行的，而程序又可按段实现保护。这种管理方式兼有两者的优点，但地址变换速度比较慢。

6. 外存储器

外存储器用来存放暂时不用的程序和数据，并且以文件的形式存储。CPU 不能直接访问外存中的程序和数据，将其以文件为单位调入主存后方可访问。外存由磁表面存储器(如磁盘、磁带)及光盘存储器构成。

1) 磁盘存储器

(1) 磁盘存储器的构成。磁盘存储器由盘片、驱动器、控制器和接口组成。盘片用来存储信息；驱动器用于驱动磁头沿盘面径向运动以寻找目标磁道位置，驱动盘片以额定速率稳定旋转，并且控制数据的写入和读出；控制器接收主机发来的命令，将它转换成磁盘驱动器的控制命令，并实现主机和驱动器之间数据格式的转换及数据传送，以控制驱动器的读写操作；接口是主机和磁盘存储器之间的连接逻辑。

(2) 磁盘存储器的种类。根据所用材质的不同，磁盘存储器分为软盘和硬盘。

① 软盘。为了正确存储信息，将盘片划成许多同心圆，称为磁道，从外到里编号，最外一圈为 0 道，往内道号依次增加。沿径向的单位距离的磁道数称为道密度，单位为 tpi。将一个磁道沿圆周等分为若干段，每段称为一个扇段或扇区，每个扇区内可存放一个固定长度的数据块。磁道上单位距离可记录的比特数称为位密度，单位为 bpi。因为每条磁道上的扇区数相同，而每个扇区的大小又一样，所以每个磁道都记录同样多的信息。又因为里圈磁道的圆周比外圈磁道的圆周小，所以里圈磁道的位密度要比外圈磁道的位密度高。最内圈的位密度称为最大位密度。

磁盘容量有两种指标：一种是非格式化容量，它是指一个磁盘所能存储的总位数；另一种是格式化容量，它是指各扇区中数据区容量的总和。计算公式分别为：

非格式化容量=面数×(磁道数/面)×内圆周长×最大位密度

格式化容量=面数×(磁道数/面)×(扇区数/道)×(字节数/扇区)

② 硬盘。按盘片是否固定、磁头是否移动等指标，硬盘可分为移动磁头固定盘片的磁盘存储器、固定磁头的磁盘存储器、移动磁头可换盘片的磁盘存储器和温彻斯特磁盘存储器(简称温盘)。一个硬盘驱动器内可装多个盘片，组成盘片组，每个盘片都配有一个独立的磁头。所以记录面上相同序号的磁道构成一个圆柱面，其编号与磁道编号相同。文件存储在硬盘上时尽可能放在同一圆柱面上，或者放在相邻柱面上，这样可以缩短寻道时间。

2) 光盘存储器

(1) 光盘存储器的类型。根据性能和用途，可分为只读型光盘、只写一次型光盘和可擦除型光盘。

(2) 光盘存储器的组成及特点。光盘存储器由光学、电学和机械部件等组成。特点是：记录密度高；存储容量大；采用非接触式读写信息；信息可长期保存；采用多通道记录时数据传输率可超过 200Mb/s；制造成本低；对机械结构的精度要求不高；存取时间较长。

7. 磁盘阵列技术

磁盘阵列是由多台磁盘存储器组成的、快速大容量且高可靠的外存子系统。现在常见的廉价冗余磁盘阵列(Redundant Array of Inexpensive Disks，RAID)，就是一种由多块廉价磁盘构成的冗余阵列。虽然 RAID 包含多块磁盘，但是在操作系统下是作为一个独立的大型存储设备出现的。RAID 技术分为几种不同的等级，分别可以提供不同的速度、安全性和性价比，如表 1-1 所示。

表 1-1 廉价冗余磁盘阵列(RAID)

RAID 等级	说 明
RAID-0	RAID-0 是一种不具备容错能力的阵列
RAID-1	RAID-1 是采用镜像容错技术改善可靠性的一种磁盘阵列
RAID-2	RAID-2 是采用海明码进行错误检测的一种磁盘阵列
RAID-3	RAID-3 减少了用于检验的磁盘存储器的台数，从而提高了磁盘阵列的有效容量。一般只有一个检验盘
RAID-4	RAID-4 是一种可独立地对组内各磁盘进行读写的磁盘阵列，该阵列也只用一个检验盘
RAID-5	RAID-5 是对 RAID-4 的一种改进，它不设置专门的检验盘。同一台磁盘上既记录数据，也记录检验信息。这就解决了前面多台磁盘机争用一台检验盘的问题
RAID-6	RAID-6 磁盘阵列采用两级数据冗余和新的数据编码以解决数据恢复问题，在两个磁盘出现故障时仍然能够正常工作。在进行写操作时，RAID-6 分别进行两个独立的校验运算，形成两个独立的冗余数据，并写入两个不同的磁盘

1.2.1.3 输入输出技术

1. 常见的内存与接口的编址方式

1) 内存与接口地址独立的编址方法

内存地址与接口地址完全独立且相互隔离，在使用中内存用于存放程序和数据，而接口就用于寻址外设。在编程序和读程序时很容易使用和辨认。缺点就是用于接口的指令太

少，功能太弱。

2)　内存与接口地址统一编址的方法

内存地址与接口地址统一在一个公共的地址空间，在这些地址空间里拿一些地址分配给接口使用而剩下的就可以归内存使用。优点是原则上用于内存的指令全部都可以用于接口；缺点就在于整个地址空间被分成两部分，一部分分配给接口使用，另一部分分配给内存使用，这经常会导致内存地址不连续。再就是用于内存的指令和用于接口的指令是完全一样的，这在读程序时就要根据参数定义表仔细加以辨认。

2.　CPU 与外设之间的数据传送方式

CPU 与外设之间的数据传送方式有以下几种。

(1)　直接程序控制。这种方式是指在完成数据的输入输出中，整个数据输出过程是在 CPU 执行程序的控制下完成的。这种方式还可以分为以下几种。

① 无条件传送方式。无条件地与 CPU 交换数据。

② 程序查询方式。先通过 CPU 查询外设状态，准备好之后再与 CPU 交换数据。程序查询方式有两大缺点。降低了 CPU 的效率；对外部的突发事件无法作出实时响应。优点在于这种思想很容易理解，同时实现这种方式工作也很容易。

(2)　中断控制。这种方式利用中断机制，当 I/O 系统外设交换数据时，CPU 无须等待，也不必查询 I/O 状态即可以抽身出来处理其他任务，因此提高了系统效率。

中断处理方法有多中断信号线法、中断软件查询法、菊花链法、总线仲裁法及中断向量表法。

(3)　中断优先级控制。这种方式分两种情况。当不同优先级的多个中断源同时提出中断请求时，CPU 应优先响应优先级最高的中断源。当 CPU 正在对某一个中断源服务时，又有比它优先级更高的中断源提出中断请求，CPU 应能暂时中断正在执行的中断服务程序而转去对优先级更高的中断源服务，服务结束后再回到原先被中断的优先级较低的中断服务程序继续执行。这种情况称为中断嵌套，即一个中断服务程序中嵌套着另一个中断服务程序。

(4)　直接存取方式。这种方式是在存储器与 I/O 设备间直接传送数据，即在内存与 I/O 设备之间传送一个数据块的过程中，不需要 CPU 的任何干涉，是一种完全由 DMA 硬件完成 I/O 操作的方式。

3.　输入输出处理机

输入输出处理机(IOP)是一个专用处理机，用于完成主机的输入输出操作。IOP 根据主机的 I/O 命令，完成对外设数据的输入和输出。它的数据传送方法有 3 种，即字节多路方式、选择传送方式和数组多路方式。

1.2.1.4　总线结构

1.　总线的定义与分类

广义地讲，任何连接两个以上电子元器件的导线都可以称为总线。通常分为以下 3 类总线。

(1)　内部总线。用于芯片一级的互连，分为芯片内总线和元件级总线。芯片内总线用

于在集成电路芯片内部各部分的连接，元件级总线用于一块电路板内各元器件的连接。

(2) 系统总线。用于插件板一级的互连，用于构成计算机各组成部分(CPU、内存和接口等)的连接。

(3) 外部总线。又称通信总线，用于设备一级的互连，通过该总线和其他设备进行信息与数据的交换。

2. 系统总线

系统总线有时也称为内总线，其性能直接影响计算机的性能。常见的内总线标准有以下 3 种。

(1) ISA(Industry Standard Architecture)总线。它是工业标准总线，向上兼容更早的 PC 总线，在 PC 总线 62 个插座信号的基础上，再扩充另一个 36 个信号的插座构成 ISA 总线。它主要包括 24 个地址线、16 条数据线等。

(2) EISA(Extended Industry Standard Architecture)总线。它是在 ISA 总线的基础上发展起来的 36 位总线。该总线定义 32 位地址线、32 位数据线以及其他控制信号线、电源线等共 196 个连接点。总线传输速率达 33Mb/s。该总线利用总线插座与 ISA 总线相兼容。

(3) PCI(Peripheral Component Interconnection，外部设备组件互连)总线。当前最流行的总线之一，它是由 Intel 公司推出的一种局部总线。它定义了 32 位数据总线，且可扩展为 64 位。PCI 总线的传输速率至少为 133Mb/s，64 位 PCI 总线的传输速率为 266Mb/s。PCI 总线的工作与处理器相互独立。PCI 总线上的设备是即插即用的。

3. 外部总线

外部总线有时简称为外总线。外总线的标准有七八十种之多，常见的外总线标准有以下几种。

(1) RS-232C。一条串行外总线，其主要特点是，所需传输线比较少，最少只需 3 条线即可实现全双工通信。传输距离远，用电平传送为 15m，电流环传送为 1km。有多种可供选择的传输速率，具有较好的抗干扰性。

(2) SCSI(Small Computer Standard Interface，小型计算机标准接口)总线。一条并行外总线，广泛用于连接软盘、光盘等。该接口早期是 8 位的，后来发展成 16 位。传输速率为 5～16Mb/s。目前的传输速率已高达 320Mb/s。该总线上最多可接 63 种外设，传输距离可达 20m。

(3) USB(Universal Serial Bus，通用串行总线)总线。USB 是 1995 年由 Microsoft、Compaq、IBM 等公司联合制定的一种新的 PC 串行通信协议。由 4 条信号线组成，可以经过集线器进行树状连接，最多可达 5 层。该总线上可接 127 个设备。最大的优点在于它支持即插即用技术并支持热插拔。

(4) IEEE-1394。一种串行外总线，由 6 条信号线组成，可接 63 个设备，传输速率从 400Mb/s、800Mb/s、1600Mb/s 直到 3.2Gb/s，最大优点在于支持即插即用并支持热插拔。

1.2.2 典型例题分析

例 1 计算机中 CPU 的中断响应时间指的是__(4)__的时间。(2015 年上半年试题 4)

(4) A．从发出中断请求到中断处理结束
B．从中断处理开始到中断处理结束
C．CPU 分析判断中断请求
D．从发出中断请求到开始进入中断处理程序

解析：中断响应时间是指计算机接收到中断信号到操作系统作出响应，并完成切换转入中断服务程序的时间。广义上的中断响应时间是指，从来自 CPU 内部或外部的中断信号发生的时刻，到 CPU 完成当前现场保存，而进入此中断信号对应的处理程序的入口处的时刻所经历的时间。

答案：D

例 2　为了便于实现多级中断嵌套，使用 (3) 来保护断电和现场最有效。(2013 年上半年试题 3)

(3) A. ROM　　B. 中断向量表　　C. 通用寄存器　　D. 堆栈

解析：多重中断时，每次中断出现的断点都必须保存起来。中断系统对断点的保存都是在中断周期内由中断隐指令实现的，对用户是透明的。断点可以保存在堆栈中，由于堆栈先进后出的特点，依次将程序的断点压入堆栈中。出栈时，按相反顺序便可准确返回到程序间断处。

答案：D

例 3　DMA 工作方式下，在 (4) 之间建立了直接的数据通路。(2013 年上半年试题 4)

(4) A. CPU 与外设　　B. CPU 与主存
C. 主存与外设　　D. 外设与外设

解析：在 DMA 模式下，CPU 只需向 DMA 控制器下达指令，让 DMA 控制器来处理数的传送，数据传送完毕再把信息反馈给 CPU，这样就很大程度上减轻了 CPU 资源占有率。DMA 请求仅仅使 CPU 暂停一下，不需要对断点和现场的处理，并且是由 DMA 控制外设与主存之间的数据传送，无需 CPU 的干预，因此 DMA 工作方式下，在主存与外设之间建立了直接的数据通路。

答案：C

例 4　地址编号为 80000H～BFFFFH 且按字节编址的内存容量为 (5) KB，若用 16K×4bit 的存储器芯片构成该内存，共需 (6) 片。(2013 年上半年试题 5、6)

(5) A. 128　　B. 256　　C. 512　　D. 1024
(6) A. 8　　B. 16　　C. 32　　D. 64

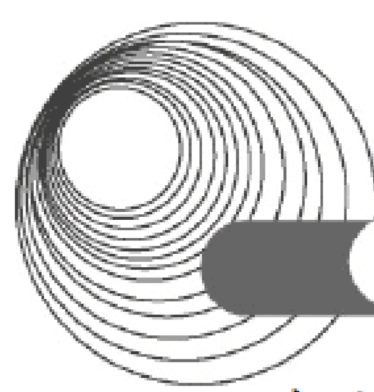
解析：从 80000H～BFFFFH 有 BFFFFH-80000H+1 个地址单元，即 40000H 个地址单元。若按字节编址，有 2^{18} 个字节，即 256KB。若用 16K×4bit 的存储芯片，需要 (256K×2×4bit)/(16K×4bit)=32 片芯片。

答案：(5) B　(6) C

例 5　在机器指令的地址字段中，直接指出操作数本身的寻址方式称为 (4) 。(2015 年下半年试题 4)

(4) A．隐含寻址　　B．寄存器寻址
C．立即寻址　　D．直接寻址

解析：隐含寻址：这种类型的指令，不是明显地给出操作数的地址；而是在指令中隐



含着操作数的地址。

寄存器寻址：当操作数不放在内存中，而是放在 CPU 的通用寄存器中时，可采用寄存器寻址方式。显然，此时指令中给出的操作数地址不是内存的地址单元号，而是通用寄存器的编号。

立即寻址：指令的地址字段指出的不是操作数的地址，而是操作数本身。立即寻址方式的特点是指令执行时间很短，因为它不需要访问内存取数，从而节省了访问内存的时间。

直接寻址是一种基本的寻址方法，其特点是：在指令格式的地址字段中直接指出操作数在内存的地址。由于操作数的地址直接给出而不需要经过某种变换，所以称这种寻址方式为直接寻址方式。

答案：C

例 6 CPU 是在(1)结束时响应 DMA 请求的。(2015 年下半年试题 1)

(1) A．一条指令执行　　B．一段程序
C．一个时钟周期　　D．一个总线周期

解析：外设向 DMA 控制器(DMAC)提出 DMA 传送的请求；然后 DMA 控制器向 CPU 提出请求;CPU 在完成当前的总线周期后立即对此请求作出相应。总线周期通常指的是 CPU 完成一次访问存储器或 I/O 端口操作所需要的时间。

答案：(1)D

例 7 (6) 不属于系统总线。(2012 年下半年试题 6)

(6) A．ISA　　B．EISA　　C．SCSI　　D．PCI

解析：系统总线又称为内总线，其速度对系统性能有极大的影响。常见的系统总线有 ISA 总线、EISA 总线、PCI 总线。SCSI 总线是并行外总线，广泛用于连接软硬磁盘、光盘和扫描仪等。

答案：C

例 8 内存按字节编址从 B3000H 到 DABFFH 的区域，其存储容量为 (5) 。(2015 年下半年试题 5)

(5) A．123KB　　B．159KB　　C．163KB　　D．194KB

解析：存储地址从 B3000H 到 DABFFH 共有(DABFFH−B3000H+1=27C00H=159K)个存储单元，由于内存地址按字节编址，所以存储容量为 159KB。

答案：B

例 9 虚拟存储体系由 (2) 两级存储器构成。(2015 年下半年试题 2)

(2) A．主存-辅存　　B．寄存器-Cache
C．寄存器-主存　　D．Cache-主存

解析：虚拟存储器采用“主存-辅存”结构。程序(数据)被分成很多小块，全部存储在辅存。运行时，需要把要用到的块先调入主存，并把马上要用到的块从主存调入高速缓存。

答案：A

例 10 若 CPU 要执行的指令为 MOV R1,#45(即将数值 45 传送到寄存器 R1 中)，则该指令中采用的寻址方式为 (4) 。(2012 年上半年试题 4)

(4) A．直接寻址和立即寻址　　B．寄存器寻址和立即寻址
C．相对寻址和直接寻址　　D．寄存器间接寻址和直接寻址

解析：操作数作为指令的一部分而直接写在指令中为立即寻址。本题中数值 45 直接包含在指令中，为立即寻址。把目标操作数存入寄存器的为寄存器寻址，本题中，需要将数值 45 传送到寄存器 R1 中，为寄存器寻址。

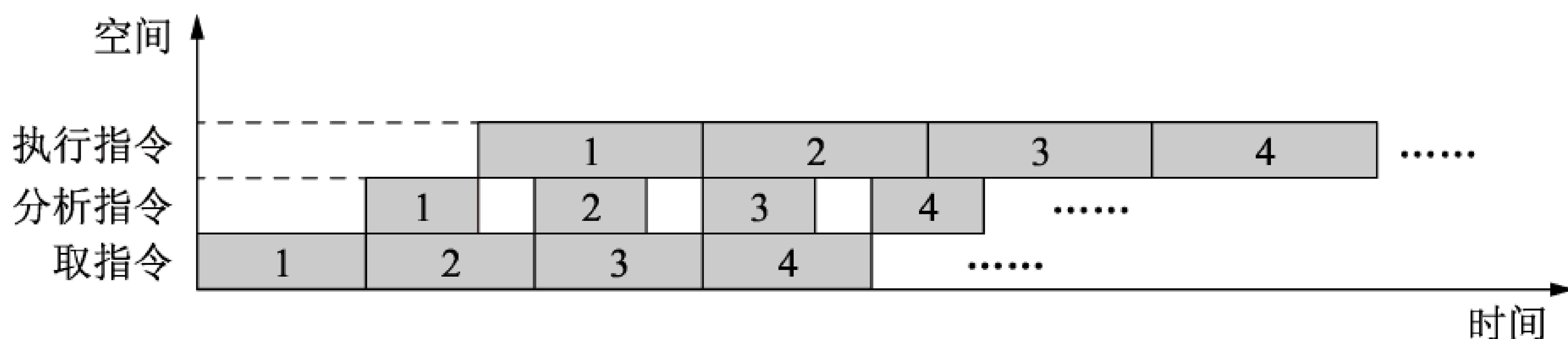
答案：B

例 11　一条指令的执行过程可以分解为取指、分析和执行三步，在取指时间 $t_{取指}=3\Delta t$、分析时间 $t_{分析}=2\Delta t$、执行时间 $t_{执行}=4\Delta t$ 的情况下，若按串行方式执行，则 10 条指令全部执行完需要 (5) Δt。若按照流水方式执行，则执行完 10 条指令需要 (6) Δt。(2012 年上半年试题 5、6)

(5) A. 40　　B. 70　　C. 90　　D. 100

(6) A. 20　　B. 30　　C. 40　　D. 45

解析：执行一条指令的时间为取指时间、分析时间、执行时间之和，为 $9\Delta t$。按串行方式执行，执行完一条指令之后才能执行下一条指令，因此执行 10 条指令的时间为 $9\Delta t\times 10=90\Delta t$。按流水方式执行的过程如下，由于执行指令的时间最长，执行 10 条指令的时间为 $10t_{执行}+t_{取指}+t_{分析}=4\Delta t\times 10+3\Delta t+2\Delta t=45\Delta t$。



答案：(5) C　　(6) D

例 12　若某计算机系统的 I/O 接口与主存采用统一编址，则输入输出操作是通过 (2) 指令来完成的。(2011 年下半年试题 2)

(2) A. 控制　　B. 中断　　C. 输入输出　　D. 访存

解析：统一编址方式是指 I/O 端口与主存单元使用同一个地址空间进行统一编址。在这种方式下，CPU 指令系统中无须设置专门的与设备进行数据传输的输入输出指令，I/O 端口被当成主存单元同样对待，对主存单元进行访问和操作的指令可以同样用于对 I/O 端口的访问和操作。

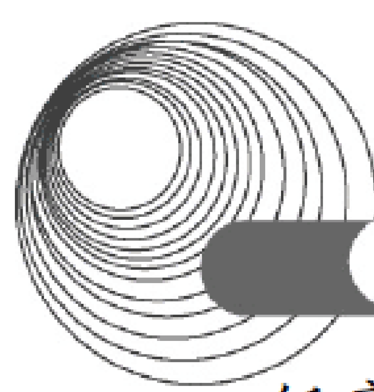
答案：D

例 13　Cache 的地址映像方式中，发生块冲突次数最小的是 (3) 。(2015 年上半年试题 3)

(3) A. 全相联映像　　B. 组相联映像　　C. 直接映像　　D. 无法确定的

解析：全相联映像：主存中任何一个块均可以映像装入到 Cache 中的任何一个块的位置上。主存地址分为块号和块内地址两部分，Cache 地址也分为块号和块内地址。Cache 的块内地址部分直接取自主存地址的块内地址段。主存块号和 Cache 块号不相同，Cache 块号根据主存块号从块表中查找。Cache 保存的各数据块互不相关，Cache 必须对每个块和块自身的地址加以存储。当请求数据时，Cache 控制器要把请求地址同所有的地址加以比较，进行确认。特点：灵活，块冲突率低，只有在 Cache 中的块全部装满后才会出现冲突，Cache 利用率高；但地址变换机构复杂，地址变换速度慢，成本高。

直接映像：把主存分成若干区，每区与 Cache 大小相同。区内分块，主存每个区中块的大小和 Cache 中块的大小相等，主存中每个区包含的块的个数与 Cache 中块的个数相等。



任意一个主存块只能映像到 Cache 中唯一指定的块中，即相同块号的位置。特点：地址变换简单、速度快，可直接由主存地址提取出 Cache 地址。但不灵活，块冲突率较高，Cache 空间得不到充分利用。

组相联映像是前两种方式的折衷。主存按 Cache 容量分区，每个区分为若干组，每组包含若干块。Cache 也进行同样的分组和分块。主存中一个组内的块数与 Cache 中一个组内的块数相等。组间采用直接方式，组内采用全相联方式。

答案：A

例 14 总线复用方式可以<u>(4)</u>。(2011 年下半年试题 4)

(4) A. 提高总线的传输宽度　　B. 增加总线的功能
C. 减少总线中信号线的数量　　D. 提高 CPU 的利用率

解析：总线复用指的是数据和地址在同一个总线上传输的方式。由于 CPU 的引脚数量有限，使得一些引脚起多个作用。采用总线复用方式，可以减少总线中信号线的数量。

答案：C

例 15 指令系统中采用不同寻址方式的目的是<u>(2)</u>。(2011 年上半年试题 2)

(2) A．提高从内存获取数据的速度　　B．提高从外存获取数据的速度
C．降低操作码的译码难度　　D．扩大寻址空间并提高编程灵活性

解析：在程序运行的过程中，形成指令地址或操作数地址的方式，称为寻址方式。寻址方式是指令系统设计的重要内容，丰富多样的寻址手段在方便程序编制、提高程序质量、减少访问主存次数以及压缩程序空间等方面起着重要作用。

答案：D

例 16 Flynn 分类法基于信息流特征将计算机分成 4 类，其中<u>(6)</u>只有理论意义而无实例。(2014 年下半年试题 6)

(6) A．SISD　　B．MISD　　C．SIMD　　D．MIMD

解析：按照 Flynn 分类法，根据计算机中指令和数据的并行状况可把计算机分成：

单指令流单数据流(SISD)——传统的计算机包含单个 CPU，它从存储在内存中的程序那里获得指令，并作用于单一的数据流。

单指令流多数据流(SIMD)——单个的指令流作用于多于一个的数据流上。例如，有数据 4、5 和 3、2，一个单指令执行两个独立的加法运算：4+5 和 3+2，就被称为单指令流多数据流。SIMD 的一个例子就是一个数组或向量处理系统，它可以对不同的数据并行执行相同的操作。

多指令流单数据流(MISD)——用多个指令作用于单个数据流的情况实际上很少见。这种冗余多用于容错系统。

多指令流多数据流(MIMD)——这种系统类似于多个 SISD 系统。实际上，MIMD 系统的一个常见例子是多处理器计算机，如 Sun 的企业级服务器。

答案：(6)　B

例 17 在输入输出控制方法中，采用<u>(1)</u>可以使得设备与主存间的数据块传送无需 CPU 干预。(2010 年下半年试题 1)

(1) A．程序控制输入输出　　B．中断　　C．DMA　　D．总线控制

解析：DMA(Direct Memory Access，直接存储器存取)技术通过硬件控制将数据块在内存和输入输出设备间直接传送，不需要 CPU 的任何干涉，只需 CPU 在过程开始启动与过程结束时的处理，实际操作由 DMA 硬件直接执行完成，CPU 在传送过程中可做别的事情。

答案： C

例 18　若内存容量为 4GB，字长为 32，则__(3)__。(2010 年下半年试题 3)

(3) A．地址总线和数据总线的宽度都为 32
　　B．地址总线的宽度为 30，数据总线的宽度为 32
　　C．地址总线的宽度为 30，数据总线的宽度为 8
　　D．地址总线的宽度为 32，数据总线的宽度为 8

解析： 在同一时间处理二进制数的位数叫字长。32 位 CPU 就是在同一时间内可处理字长为 32 位的二进制数据。地址总线的宽度决定了内存容量，如果地址总线宽度为 32，则存储容量为 2^{32}=4GB。

答案： A

例 19　内存按字节编址从 A5000H 到 DCFFFH 的区域，其存储容量为__(4)__。(2014 年下半年试题 4)

(4) A．123KB　　B．180KB　　C．223KB　　D．224KB

解析： DCFFFH−A5000H+1H=38000H=224KB

答案： D

例 20　在 CPU 中，常用来为 ALU 执行算术逻辑运算提供数据并暂存运算结果的寄存器是__(1)__。(2014 年上半年试题 1)

(1) A. 程序计数器　B. 状态寄存器　C. 通用寄存器　D. 累加寄存器

解析： 在运算器中，累加寄存器是专门存放算术或逻辑运算的一个操作数和运算结果的寄存器，能进行加、减、读出、移位、循环移位和求补等操作，是运算器的主要部分。

答案： D

例 21　以下关于计算机系统中断概念的叙述中，正确的是__(3)__。(2010 年上半年试题 3)

(3) A．由 I/O 设备提出的中断请求和电源掉电都是可屏蔽中断
　　B．由 I/O 设备提出的中断请求和电源掉电都是不可屏蔽中断
　　C．由 I/O 设备提出的中断请求是可屏蔽中断，电源掉电是不可屏蔽中断
　　D．由 I/O 设备提出的中断请求是不可屏蔽中断，电源掉电是可屏蔽中断

解析： 按照是否可以被屏蔽，可将中断分为两大类：不可屏蔽中断(又叫非屏蔽中断)和可屏蔽中断。不可屏蔽中断源一旦提出请求，CPU 必须无条件响应，而对可屏蔽中断源的请求，CPU 可以响应也可以不响应。典型的非屏蔽中断源的例子是电源掉电，一旦出现，必须立即无条件地响应；否则进行其他任何工作都是没有意义的。典型的可屏蔽中断源的例子是打印机中断，CPU 对打印机中断请求的响应可以快一些，也可以慢一些，因为让打印机等待是完全可以的。

答案： C

例 22　计算机指令一般包括操作码和地址码两部分，为分析执行一条指令，其__(5)__。(2010 年上半年试题 5)

(5) A．操作码应存入指令寄存器(IR)，地址码应存入程序计数器(PC)
　　B．操作码应存入程序计数器(PC)，地址码应存入指令寄存器(IR)
　　C．操作码和地址码都应存入指令寄存器(IR)
　　D．操作码和地址码都应存入程序计数器(PC)

解析：程序被加载到内存后开始运行，当CPU执行一条指令时，先把它从内存储器取到缓冲寄存器DR中，再送入IR暂存，指令译码器根据IR的内容产生各种微操作指令，控制其他的组成部件工作，完成所需的功能。

答案：C

例23 假设某磁盘的每个磁道划分成9个物理块，每块存放一个逻辑记录。逻辑记录R0，R1，…，R8存放在同一个磁道，记录的安排顺序如下表所示。

物理块	1	2	3	4	5	6	7	8	9
逻辑记录	R0	R1	R2	R3	R4	R5	R6	R7	R8

如果磁盘的旋转速度为27ms/周，磁头当前处在R0的开始处。若系统顺序处理这些记录，使用单缓冲区，每个记录处理时间为3ms，则处理这9个记录的最长时间为<u>(27)</u>；若对信息存储进行优化分布后，处理9个记录的最少时间为<u>(28)</u>。(2010上半年试题27、28)

(27) A．54ms　　B．108ms　　C．222ms　　D．243ms

(28) A．27ms　　B．54ms　　C．108ms　　D．216ms

解析：系统读记录的时间为27/9=3ms。对第一种情况：系统读出并处理记录R1之后，将转到记录R3的开始处，所以为了读出记录R2，磁盘必须再转一圈，需要27ms(转一圈)的时间。这样，处理9个记录的总时间应为222ms。因为处理前8个记录(即R1，R2，…，R8)的总时间再加上读R9时间：8×27ms+6ms=222ms。

对于第二种情况，若对信息进行分步优化的结果如下所示。

物理块	1	2	3	4	5	6	7	8	9
逻辑记录	R1	R6	R2	R7	R3	R8	R4	R9	R5

从上表可以看出，当读出记录R1并处理结束后，磁头刚好转至R2记录的开始处，立即就可以读出并处理，因此处理9个记录的总时间为

9×(3ms(读记录)+3ms(处理记录))=9×6ms=54ms

答案：(27) C　(28) B

例24 以下关于RISC和CISC的叙述中，不正确的是<u>(6)</u>。(2015年上半年试题6)

(6) A．RISC通常比CISC的指令系统更复杂

B．RISC通常会比CISC配置更多的寄存器

C．RISC编译器的子程序库通常要比CISC编译器的子程序库大得多

D．RISC比CISC更加适合VLSI工艺的规整性要求

解析：异步流水线是指，流水线输出端任务流出的顺序与输入端任务流入的顺序可以不同，允许后进入流水线的任务先完成。这种流水线又称为乱序流水线、无序流水线、错序流水线。CPU层次的东西，不会对上层产生结果性的影响，但乱序执行与分支预测会对程序的效率产生影响，因为每一次分支判错，都会丢弃计算的结果。

答案：D

例25 以下关于Cache的叙述中，正确的是<u>(6)</u>。(2009年下半年试题6)

(6) A．在容量确定的情况下，替换算法的时间复杂度是影响Cache命中率的关键因素

B．Cache的设计思想是在合理成本下提高命中率

C．Cache 的设计目标是容量尽可能与主存容量相等

D．CPU 中的 Cache 容量应大于 CPU 之外的 Cache 容量

解析：Cache 的性能是计算机系统性能的重要方面。命中率是 Cache 的一个重要指标，但不是最主要的指标。Cache 设计的主要目标是在成本允许的情况下达到较高的命中率，使存储系统具有最短的平均访问时间。Cache 的命中率和 Cache 容量的关系是：Cache 容量越大，则命中率越高，随着容量的增加，其失效率接近 0%(命中率接近 100%)。但是，增加 Cache 的容量意味着增加 Cache 的成本和增加 Cache 的命中时间。

答案：B

例 26　假设某硬盘由 5 个盘片构成(共有 8 个记录面)，盘面有效记录区域的外直径为 30cm，内直径为 10cm，记录位密度为 250 位/mm，磁道密度为 16 道/mm，每磁道分 16 个扇区，每扇区 512B，则该硬盘的格式化容量约为<u>(2)</u>MB。(2009 年上半年试题 2)

(2) A. $\frac{8\times(30-10)\times10\times250\times16}{8\times1024\times1024}$　　B. $\frac{8\times(30-10)\times10\times16\times16\times512}{2\times1024\times1024}$

C. $\frac{8\times(30-10)\times10\times250\times16\times16}{8\times1024\times1024}$　　D. $\frac{8\times(30-10)\times16\times16\times512}{2\times1024\times1024}$

解析：磁盘容量有两种指标，一种是非格式化容量，指一个磁盘所能存储的总位数；另一种是格式化容量，指各扇区中数据区容量总和。计算公式分别为

非格式化容量=面数×(磁道数/面)×内圆周长×最大位密度

格式化容量=面数×(磁道数/面)×(扇区数/道)×(字节数/扇区)

本题目求的是格式化容量，套用第二个公式即可。

答案：B

例 27　<u>(3)</u>是指按内容访问的存储器。(2009 年上半年试题 3)

(3) A．虚拟存储器　　B．相联存储器

C．高速缓存　　D．随机访问存储器

解析：存储器按访问方式可分为按地址访问和按内容访问。相联存储器的工作原理是把数据或者数据的某一部分作为关键字，将该关键字与存储器中的每一个单元进行比较，找出存储器中所有与关键字相同的数据字。显然，相联存储器是按内容访问的存储器。其他存储器都是按地址访问的。

答案：B

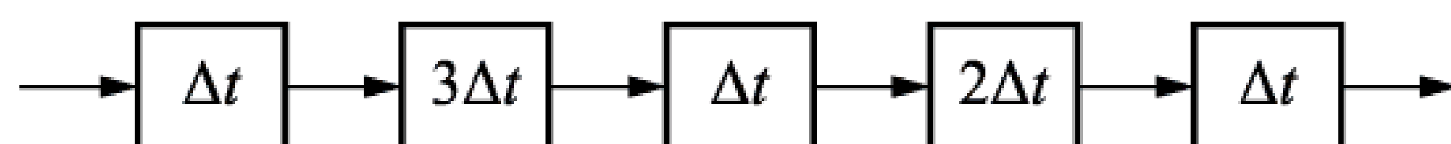
例 28　总线宽度为 32bit，时钟频率为 200MHz，若总线上每 5 个时钟周期传送一个 32bit 的字，则该总线的带宽为<u>(5)</u>MB/s。(2015 年上半年试题 5)

(5)　A．40　　B．80　　C．160　　D．200

解析：频率为 200MHz，每 5 个时间周期传一个字，则 1s 内可传送 200M/5=40M 次，每次 32bit，4 个字节，得出总线带宽为 40M × 4=160MB/s。

答案：C

例 29　某指令的流水线由 5 段组成，第 1、3、5 段所需时间为Δt，第 2、4 段所需时间分别为 $3\Delta t$、$2\Delta t$，如下图所示，那么连续输入 n 条指令时的吞吐率(单位时间内执行的指令个数)TP 为<u>(6)</u>。(2009 年上半年试题 6)



(6) A. $\frac{n}{5\times(3+2)\Delta t}$　　B. $\frac{n}{(3+3+2)\Delta t+3(n-1)\Delta t}$

C. $\frac{n}{(3+2)\Delta t+(n-3)\Delta t}$　　D. $\frac{n}{(3+2)\Delta t+5\times3\Delta t}$

解析：吞吐率是指单位时间里流水线处理机流出的结果数。对指令而言即为单位时间里执行的指令数。对于这一题，流水线的子过程所用的时间不同，所以指令第一次执行时间应该为(1+3+1+2+1)Δt，从第二次开始，指令在流水操作中的时间应该看最长子过程所用的时间，一共有 n−1 次，所以总时间为(1+3+1+2+1)Δt+3(n−1)Δt。本题中连续输入 n 条指令，所以完成这 n 个任务所需的时间为 T=(1+3+1+2+1)Δt+3(n−1)Δt，故吞吐率为 n/T。

答案：B

1.2.3　同步练习

1.　在 CPU 与主存之间设置高速缓冲存储器(Cache)，其目的是为了____。

A．扩大主存的存储容量　　B．提高 CPU 对主存的访问效率

C．既扩大主存容量又提高存取速度　　D．提高外存储器的速度

2.　下面的描述中，____不是 RISC 设计应遵循的设计原则。

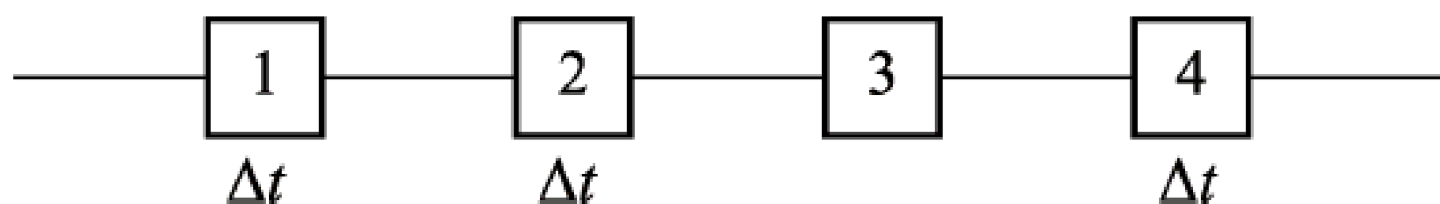
A．指令条数应少一些

B．寻址方式尽可能少

C．采用变长指令，功能复杂的指令长度长而简单指令长度短

D．设计尽可能多的通用寄存器

3.　指令流水线将一条指令的执行过程分为四步，其中第 1、2 和 4 步的经过时间为Δt，如下图所示。若该流水线顺序执行 50 条指令共用 153Δt，并且不考虑相关问题，则该流水线的第 3 步的时间为____Δt。



A．2　　B．3　　C．4　　D．5

4.　若内存按字节编址，用存储容量为 32K×8bit 的存储器芯片构成地址编号 A0000H～DFFFFH 的内存空间，则至少需要____片。

A．4　　B．6　　C．8　　D．10

5.　设指令由取指、分析、执行 3 个子部件完成，每个子部件的工作周期均为Δt，采用常规标量单流水线处理机。若连续执行 10 条指令，则共需时间____Δt。

A．8　　B．10　　C．12　　D．14

6.　某计算机指令字长为 16 位，指令有双操作数、单操作数和无操作数 3 种格式，每个操作数字段均用 6 位二进制表示，该指令系统共有 m 条(m<16)双操作数指令，并存在无操作数指令。若采用扩展操作码技术，那么最多还可设计出_____条单操作数指令。

A．2^6　　B．$(2^4-m)\times2^6-1$　　C．$(2^4-m)\times2^6$　　D．$(2^4-m)\times(2^6-1)$

7.　计算机内存一般分为静态数据区、代码区、栈区和堆区，若某指令的操作数之一采用立即数寻址方式，则该操作数位于____。

A．静态数据区　B．代码区　C．栈区　D．堆区

8．下面关于在 I/O 设备与主机间交换数据的叙述，____是错误的。

A．中断方式下，CPU 需要执行程序来实现数据传送任务

B．中断方式和 DMA 方式下，CPU 与 I/O 设备都可同步工作

C．中断方式和 DMA 方式中，快速 I/O 设备更适合采用中断方式传递数据

D．若同时接到 DMA 请求和中断请求，CPU 优先响应 DMA 请求

9．Cache 用于存放主存数据的部分副本，主存单元地址与 Cache 单元地址之间的转换工作由____完成。

A．硬件　B．软件　C．用户　D．程序员

10．内存按字节编址，地址从 90000H 到 CFFFFH，若用存储容量为 16K×8bit 的存储器芯片构成该内存，至少需要____片。

A．2　B．4　C．8　D．16

11．有四级指令流水线，分别完成取指、取数、运算、传送结果四步操作。若完成上述操作的时间依次为 9ns、10ns、6ns、8ns，则流水线的操作周期应设计为____ns。

A．6　B．8　C．9　D．10

12．CPU 中的数据总线宽度会影响____。

A．内存容量的大小　B．系统的运算速度

C．指令系统指令数量　D．寄存器的宽度

13．利用高速通信网络将多台高性能工作站或微型机互连构成机群系统，其系统结构形式属于____计算机。

A．单指令流单数据流(SISD)　B．多指令流单数据流(MISD)

C．单指令流多数据流(SIMD)　D．多指令流多数据流(MIMD)

14．内存采用段式存储管理有许多优点，但“____”不是其优点。

A．分段是信息的逻辑单位，用户不可见

B．各段程序的修改互不影响

C．地址变换速度快，内存碎片少

D．便于多道程序共享主存的某些段

15．在指令系统的各种寻址方式中，获取操作数最快的方式是(1)。若操作数的地址包含在指令中，则属于 (2) 方式。

(1)～(2)　A．直接寻址　B．立即寻址

C．寄存器寻址　D．间接寻址

16．若每一条指令都可以分解为取指、分析和执行三步。已知取指时间 $t_{取指}=4\Delta t$，分析时间 $t_{分析}=3\Delta t$，执行时间 $t_{执行}=5\Delta t$。如果按串行方式执行完 100 条指令需要(1) Δt。如果按照流水方式执行，执行完 100 条指令需要(2) Δt。

(1)　A．1190　B．1195　C．1200　D．1205

(2)　A．504　B．507　C．508　D．510

17．若内存地址区间为 4000H～43FFH，每个存储单元可存储 16 位二进制数，该内存区域用 4 片存储器芯片构成，则构成该内存所用的存储器芯片的容量是____。

A．512×16bit　B．256×8bit　C．256×16bit　D．1024×8bit

1.2.4 同步练习参考答案

1．B　　2．C
3．B　　4．C
5．C　　6．B
7．B　　8．C
9．A　　10．D
11．D　　12．B
13．D　　14．C
15．(1) B　(2) A　　16．(1) C　(2) B
17．C

1.3 安全性、可靠性与系统评测基础知识

1.3.1 考点辅导

1.3.1.1 计算机安全概述

计算机安全是指计算机资源安全，是要保证这些计算机资源不受自然和人为的有害因素的威胁和危害。计算机资源由系统资源和信息资源两大部分组成。系统资源包括硬件、软件、配套设备设施、有关文件资料，还可以包括有关的服务系统和业务工作人员。信息资源包括计算机系统中存储、处理和传输的大量各种各样的信息。

1. 信息安全的基本要素

信息安全的 5 个基本要素如下。

(1) 机密性。确保信息不暴露给未授权的实体或进程。

(2) 完整性。只有得到允许才能修改数据，并能够判别出数据是否已被篡改。

(3) 可用性。得到授权的实体在需要时可访问数据。

(4) 可控性。可以控制授权范围内的信息流向及行为方式。

(5) 可审查性。对出现的安全问题提供调查的依据和手段。

2. 计算机的安全等级

计算机系统中的三类安全性是指技术安全性、管理安全性及政策法律安全性。一些重要的安全评估准则有以下几个。

- 美国国防部(DOD)和国家标准局(现更名为 NIST)的可信计算机系统评估准则。
- 欧洲共同体的信息技术安全评估准则(ITSEC)。
- ISO/IEC 国际标准。
- 美国联邦标准。

其中美国国防部和国家标准局的可信计算机系统评测标准将系统划分为 4 组 7 个等级，如表 1-2 所示。

表 1-2　安全性的级别

组	安全级别	定　义
1	A1	可验证安全设计：提供 B3 级保护的同时给出系统的形式化隐秘通道分析、非形式化代码的一致性验证
2	B3	安全域：该级的 TCB(可信计算基础)必须满足访问监控器的要求，提供系统恢复过程
	B2	结构化安全保护：建立形式化的安全策略模型，并对系统内的所有主体和客体实施自主访问和强制访问控制
	B1	标记安全保护：对系统的数据加以标记，并对标记的主体和客体实施强制存取控制
3	C2	受控访问控制：实际上是安全产品的最低档次，提供受控的存取保护，存取控制以用户为单位
	C1	只提供了非常初级的自主安全保护，能实现对用户和数据的分离，进行自主存取控制，数据的保护以用户组为单位
4	D	最低级别，保护措施很小，没有安全功能

3. 安全威胁

安全威胁是指某个人、物、事件对某一资源的机密性、完整性、可用性或合法性所造成的危害。某种攻击就是威胁的具体实现。安全威胁分为两类，即故意(如黑客渗透)或偶然(如信息发往错误的地址)。典型的安全威胁举例如表 1-3 所示。

表 1-3　典型的安全威胁

威　胁	说　明
授权侵犯	为某一特权使用一个系统的人却将该系统用作其他未授权的目的
拒绝服务	对信息或其他资源的合法访问被无条件拒绝，或推迟与时间密切相关的操作
窃听	信息从被监视的通信过程中泄露出去
信息泄露	信息被泄露或暴露给某个未授权的实体
截获/修改	某一通信数据项在传输过程中被改变/删除或替代
假冒	一个实体(人或系统)假装成另一个实体
否认	参与某次通信交换的一方否认曾发生过此次交换
非法使用	资源被某个未授权的人或者未授权的方式使用
人员疏忽	一个授权的人为了金钱或利益或由于粗心将信息泄露给未授权的人
完整性破坏	通过对数据进行未授权的创建、修改或破坏，使数据的一致性受到损坏
媒体清理	信息被从废弃的或打印过的媒体中获得
物理入侵	一个入侵者通过绕过物理控制而获得对系统的访问
资源耗尽	某一资源(如访问端口)被故意超负荷地使用，导致其他用户的服务被中断

4. 影响数据安全的因素

影响数据安全的因素包括内部因素和外部因素。

(1) 内部因素。可采用多种技术对数据加密；制定数据安全规划；建立安全存储体系；建立事故应急计划和容灾措施；重视安全管理，制定数据安全管理规范。

(2) 外部因素。可将数据分成不同的密级，规定外部使用人员的权限。设置身份认证、密码、指纹、声纹、笔记等多种认证；设置防火墙，为计算机建立一道屏障，防止外部入侵破坏数据；建立入侵检测、审计和追踪，对计算机进行防卫。同时，也包括计算机物理环境的保障、防辐射、防水、防火等外部防灾措施。

1.3.1.2 加密技术和认证技术

1. 加密技术

1) 加密技术概述

加密技术是最常用的安全保密手段，数据加密技术的关键在于加密/解密算法和密钥管理。加密技术包括两个元素，即算法和密钥。数据加密的基本过程就是对原来为明文的文件或数据按某种加密算法进行处理，使其成为不可读的一段代码，通常称为“密文”。“密文”只能在输入相应的密钥之后才能显示出原来的内容，通过这样的途径达到保护数据不被窃取。

数据加密和解密是一对逆过程。数据加密是用加密算法 E 和加密密钥 K_1，将明文 P 变换成密文 C，记为

$$C = E_{K_1}(P)$$

数据解密是数据加密的逆过程，是用解密算法 D 和解密密钥 K_2 将密文 C 变换成明文 P，记为

$$P = D_{K_2}(C)$$

数据加密技术可分成 3 类，即对称加密、非对称加密和不可逆加密。

(1) 对称加密技术。对称加密的体制模型如图 1-8 所示。

常用的对称加密算法有以下几个。

- 数据加密标准(Digital Encryption Standard，DES)算法。
- 三重 DES(3DES 或称 TDES)。
- RC-5(Rivest Cipher 5)。
- 国际数据加密算法(International Data Encryption Algorithm，IDEA)。

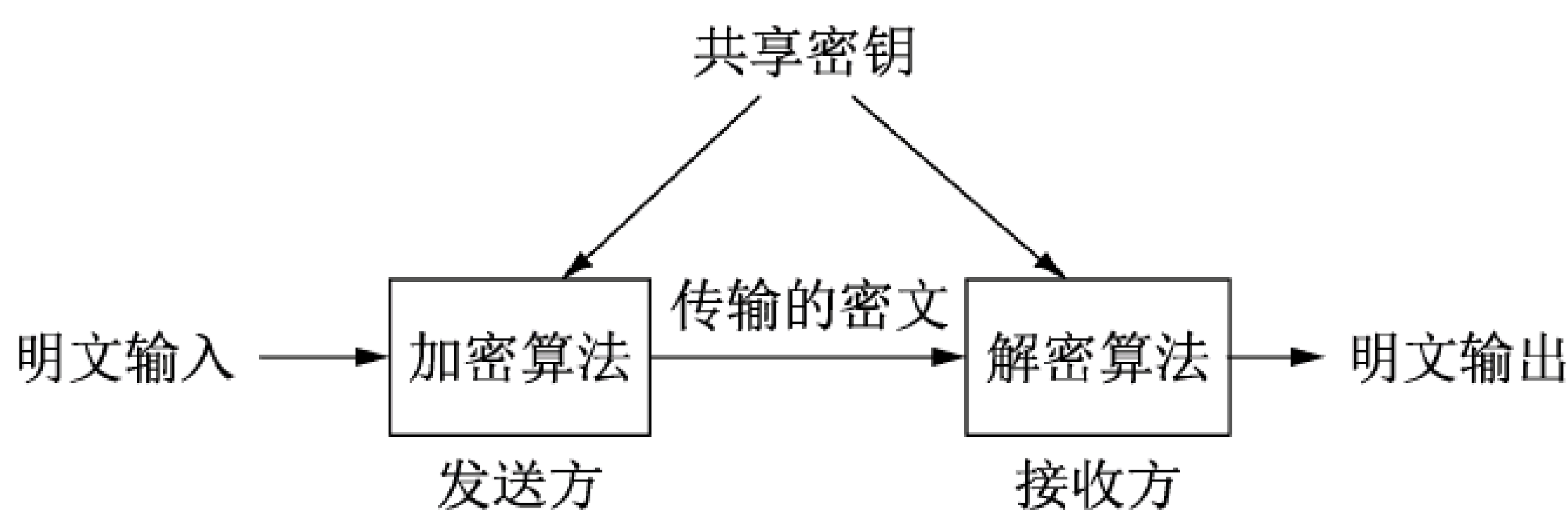


图 1-8 对称加密的体制模型

(2) 非对称加密技术。与对称加密算法不同，非对称加密算法需要两个密钥，即公开

密钥(即公钥)和私有密钥(即私钥)。这里的“公钥”是指可以对外公布的，“私钥”只能由持有人知道。公开密钥与私有密钥是一对，如果用公开密钥对数据进行加密，只有用对应的私有密钥才能解密；如果用私有密钥对数据进行加密，则只有用对应的公开密钥才能解密。因为加密和解密使用的是两个不同的密钥，所以这种算法称为非对称加密算法。

非对称加密有两个不同的体制，如图 1-9 所示。

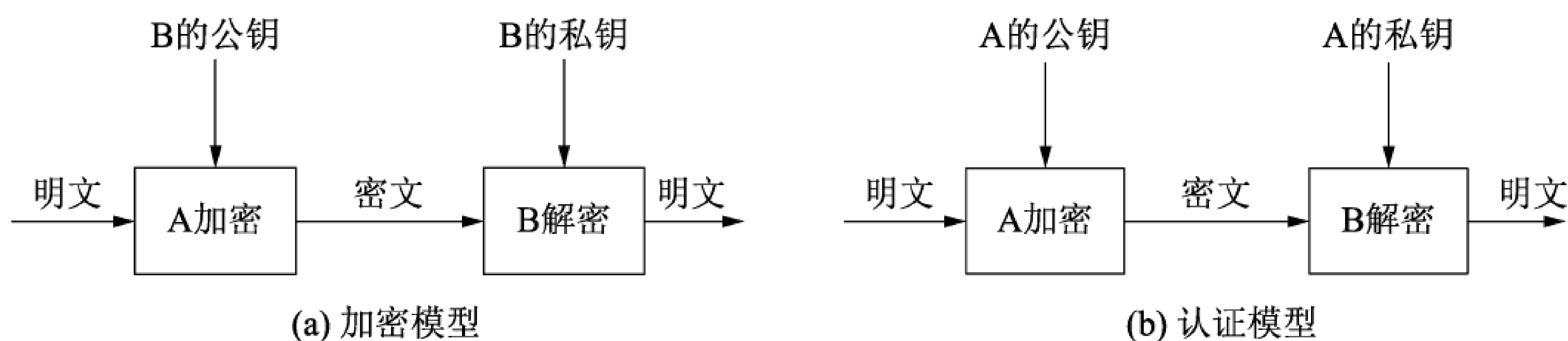


图 1-9　非对称加密的体制模型

非对称加密算法实现机密信息交换的基本过程是：甲方生成一对密钥并将其中的一把作为公用密钥向其他方公开；得到该公用密钥的乙方使用该密钥对机密信息进行加密后再发送给甲方；甲方再用自己保存的另一把专用密钥对加密后的信息进行解密。甲方只能用其专用密钥解密由其公用密钥加密后的任何信息。

非对称加密算法的保密性能好，它消除了最终用户交换密钥的需要，但加密和解密花费时间长、速度慢，不适合对文件加密，而只适用于对少量数据进行加密。

2)　密钥管理

密钥管理主要是指密钥对的安全管理，包括密钥产生、密钥备份和恢复、密钥更新以及多密钥管理。

2.　认证技术

1)　认证技术概述

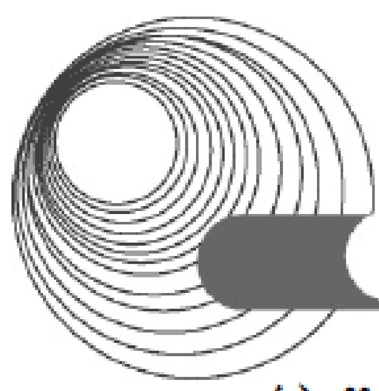
认证技术主要解决网络通信过程中通信双方的身份认证。认证的过程涉及加密和密钥交换。通常，加密可使用对称加密、不对称加密及两种加密方法的混合方法。认证一般有账户名/口令认证、使用摘要算法认证、基于 PKI(Public Key Infrastructure，公开密钥体系)的认证等几种方法。一个有效的 PKI 系统必须是安全的和透明的，用户在获得加密和数字签名服务时，不需要详细了解 PKI 的内部运行机制。

PKI 是一种遵循既定标准的密钥管理平台，它能够为所有网络应用提供加密和数字签名等密码服务及必需的密钥和证书管理体系。简单地说，PKI 是通过使用公开密钥技术和数字证书来确保系统信息安全并负责验证数字证书持有者身份的一种体系。PKI 技术是安全信息技术的核心，也是电子商务的关键和基础技术。PKI 的基础技术包括加密、数字签名、数据完整性机制、数字信封、双重数字签名等。完整的 PKI 系统必须具有权威认证机构(CA)、数字证书库、密钥备份及恢复系统、证书作废系统、应用接口(API)等基本构成部分。

2)　哈希函数与信息摘要

Hash(哈希)函数(又称散列函数)提供了这样一种计算过程：输入一个长度不固定的字符串，返回一串定长度的字符串(又称 Hash 值)。单向 Hash 函数用于产生信息摘要。

信息摘要简要地描述了一份较长的信息或文件，它可以被看作一份长文件的“数字指



纹”。信息摘要用于创建数字签名。

3) 数字签名

数字签名是通过一个单向散列函数对要传送的报文进行处理得到的，用以认证报文来源并核实报文是否发生变化的一个字母数字串。数字签名可以解决否认、伪造、篡改及冒充等问题，应用范围十分广泛，如加密信件、商务信函、订货购买系统、远程金融交易、自动模式处理等。

数字签名和数字加密的过程虽然都使用公开密钥体系，但实现的过程正好相反，使用的密钥对也不同。数字签名使用的是发送方的密钥对，发送方用自己的私有密钥进行加密，接收方用发送方的公开密钥进行解密，这是一个一对多的关系，任何拥有发送方公开密钥的人都可以验证数字签名的正确性。数字加密则使用的是接收方的密钥对，这是多对一的关系，任何知道接收方公开密钥的人都可以向接收方发送加密信息，只有唯一拥有接收方私有密钥的人才能对信息解密。另外，数字签名只采用了非对称密钥加密算法，它能保证发送信息的完整性、身份认证和不可否认性，而数字加密则采用了对称密钥加密算法和非对称密钥加密算法相结合的方法，它能保证发送信息的保密性。

4) SSL 协议

SSL(Secure Sockets Layer)是网景(Netscape)公司提出的基于 Web 应用的安全协议，又叫安全套接层协议。

SSL 协议主要提供三方面的服务：用户和服务器的合法性认证；加密数据以隐藏被传送的数据；保护数据的完整性，目的是在两个通信应用程序之间提供私密性和可靠性。对于电子商务应用来说，使用 SSL 可保证信息的真实性、完整性和保密性。

5) 数字时间戳技术

数字时间戳技术是数字签名技术的一种变种。数字时间戳服务(Digital Time-stamp Service，DTS)是网上电子商务提供的安全服务项目之一，能提供电子文件的日期和时间信息的安全保护。

时间戳(Time-stamp)是一个经加密后形成的凭证文档，它包括以下 3 个部分。

- 需加时间戳的文件的摘要。
- DTS 收到文件的日期和时间。
- DTS 的数字签名。

一般来说，时间戳产生的过程为：用户首先将需要时间戳的文件用 Hash 编码加密形成摘要，然后将该摘要发送到 DTS，DTS 在加入收到文件摘要的日期和时间信息后再对该文件加密(数字签名)，然后送回用户。

1.3.1.3 计算机可靠性

1. 计算机可靠性概述

计算机系统的硬件故障通常是由元器件的失效引起的。

计算机系统的可靠性是指从它开始运行(t=0)到某个时刻 t 这段时间内能正常运行的概率，用 $R(t)$表示。失效率是指单位时间内失效的元件数与元件总数的比例，用λ表示，当λ为常数时，可靠性与失效率的关系为 $R(t)=e^{-\lambda t}$ 。

两次故障之间系统能正常工作的时间的平均值称为平均无故障时间(MTBF)，即

MTBF=1/λ。

通常用平均修复时间(MTRF)来表示计算机的可维修性，即计算机的维修效率，指从故障发生到机器修复平均所需要的时间。计算机的可用性是指计算机的使用效率，它以系统在执行任务的任意时刻能正常工作的概率 A 来表示，即

$$A = \frac{\text{MTBF}}{\text{MTBF} + \text{MTRF}}$$

计算机的 RAS 技术是指用可靠性 R、可用性 A 和可维修性 S 这 3 个指标衡量一个计算机系统。但在实际应用中，引起计算机故障的原因除了元器件以外，还与组装工艺、逻辑设计等因素有关。

2. 计算机可靠性模型

常见的计算机系统可靠性数学模型如下。

(1) 串联系统：可靠性 $R = R_1 R_2 \cdots R_N$，失效率 $\lambda = \lambda_1 + \lambda_2 + \cdots + \lambda_N$。

(2) 并联系统：可靠性 $R = 1 - (1 - R_1) \times (1 - R_2) \times \cdots \times (1 - R_N)$，失效率 $\mu = \dfrac{1}{\dfrac{1}{\lambda}\sum\limits_{j=1}^{N}\dfrac{1}{j}}$。

(3) N 模冗余系统：$R = \sum\limits_{i=n+1}^{N} \begin{pmatrix} j \\ N \end{pmatrix} \times R_0^i (1 - R_0)^{N-i}$。

提高计算机的可靠性一般采取提高元器件质量、改进加工工艺与工艺结构、完善电路设计以及发展容错技术等措施。

1.3.1.4 计算机系统的性能评价

1. 性能评测常用方法

(1) 时钟频率。计算机的时钟频率在一定程度上反映了机器速度。一般来讲，主频越快速度越快。

(2) 指令执行速度。速度是计算机的主要性能指标之一，在计算机发展初期，曾用加法指令的运算速度来衡量计算机的速度。

(3) 等效指令速度法。统计各类指令在程序中所占的比例，并进行折算。设某类指令 i 在程序中所占比例为 ω_i，执行时间为 t_i，则等效指令时间为

$$T = \sum_{i=1}^{n} (\omega_i t_i)$$

式中 n 为指令的种类数。

(4) 数据处理速率(PDR)法。采用计算 PDR 值的方法来衡量机器性能，PDR 值越大，机器性能越好，PDR 与每条指令和每个操作数的平均位数以及每条指令的平均运算速度有关。

(5) 核心程序法。把应用程序中用得最频繁的那部分核心程序作为评价计算机性能的标准程序，在不同的机器上运行，测得其执行时间，作为各类机器性能评价的依据。

2. 基准测试程序

基准测试程序法是目前一致承认的测试性能的较好方法，有多种多样的基准程序，如主要测试整数性能的基准程序，测试浮点性能的基准程序等。常见的基准测试程序有以下

4 种。

- 整数测试程序。
- 浮点测试程序。
- SPEC 基准测试程序。
- TPC 基准程序。

1.3.2 典型例题分析

例 1 利用报文摘要算法生成报文摘要的目的是__(7)__。(2013 年上半年试题 7)

(7) A. 验证通信对方的身份，防止假冒

B. 对传输数据进行加密，防止数据被窃听

C. 防止发送方否认发送过数据

D. 防止发送的报文被篡改

解析：报文摘要是用来保证数据完整性的。传输的数据一旦被修改，那么计算出的摘要就不同，只要对比两次摘要就可确定数据是否被修改过，其目的是为了防止发送的报文被篡改。

答案：D

例 2 防火墙不具备__(8)__功能。(2015 年下半年试题 8)

(8) A. 记录访问过程　　B. 查毒　　C. 包过滤　　D. 代理

解析：防火墙指的是一个由软件和硬件设备组合而成、在内部网和外部网之间、专用网与公共网之间的构造的保护屏障，使 Internet 与 Intranet 之间建立起一个安全网关(Security Gateway)，从而保护内部网免受非法用户的侵入。防火墙技术经历了包过滤、应用网关代理和状态检测 3 个发展阶段。防火墙的目的就是在网络连接之间建立一个安全控制点，通过允许、拒绝或重新定向经过防火墙的数据流，实现对进、出内部网络的服务和访问的审计和控制。防火墙不具备查毒功能，如果系统已经中毒，防火墙是无法检测的，需要由杀毒软件完成。

答案：B

例 3 以下关于木马程序的叙述中，正确的是__(6)__。(2014 年上半年试题 6)

(6) A. 木马程序主要通过移动磁盘传播

B. 木马程序的客户端运行在攻击者的机器上

C. 木马程序的目的是使计算机或网络无法提供正常的服务

D. Sniffer 是典型的木马程序

解析：木马(Trojan)是指通过特定的程序(木马程序)来控制另一台计算机。木马通常有两个可执行程序：一个是控制端；另一个是被控制端。植入对方计算机的是服务端，而黑客正是利用客户端进入运行了服务端的计算机。运行了木马程序的服务端以后，会产生一个有着容易迷惑用户的名称的进程，暗中打开端口，向指定地点发送数据(如网络游戏的密码、即时通信软件密码和用户上网密码等)，黑客甚至可以利用这些打开的端口进入计算机系统。

答案：B

例 4　下列安全协议中，与 TLS 最接近的协议是__(7)__。(2012 年下半年试题 7)

(7) A．PGP　　B．SSL　　C．HTTPS　　D．IPSec

解析：安全传输层协议(TLS)用于在两个通信应用程序之间提供保密性和数据完整性。该协议由两层组成，即 TLS 记录协议(TLS Record)和 TLS 握手协议(TLS Handshake)。较低的层为 TLS 记录协议，位于某个可靠的传输协议(如 TCP)上面。

SSL(Secure Sockets Layer，安全套接层)及其继任者 TLS(Transport Layer Security)是为网络通信提供安全及数据完整性的一种安全协议。TLS 与 SSL 在传输层对网络连接进行加密。SSL 协议位于 TCP/IP 协议与各种应用层协议之间，为数据通信提供安全支持。SSL 协议可分为两层：① SSL 记录协议，它建立在可靠的传输协议(如 TCP)之上，为高层协议提供数据封装、压缩、加密等基本功能的支持；②SSL 握手协议，它建立在 SSL 记录协议之上，用于在实际的数据传输开始前，通信双方进行身份认证、协商加密算法、交换加密密钥等。

PGP(Pretty Good Privacy)是一个基于 RSA 公钥加密体系的邮件加密软件。可以用它对邮件保密以防止非授权者阅读，它还能对邮件加上数字签名，从而使收信人可以确认邮件的发送者，并能确信邮件没有被篡改。

HTTPS 是以安全为目标的 HTTP 通道。HTTPS 的安全基础是 SSL。

“Internet 协议安全性(IPSec)”是一种开放标准的框架结构，通过使用加密的安全服务以确保在 Internet 协议(IP)网络上进行保密而安全的通信。

答案：B

例 5　用户 B 收到用户 A 带数字签名的消息 M，为了验证 M 的真实性，首先需要从 CA 获取用户 A 的数字证书，并利用__(8)__验证该证书的真伪，然后利用__(9)__验证 M 的真实性。(2012 年下半年试题 8、9)

(8) A．CA 的公钥　　B．B 的私钥　　C．A 的公钥　　D．B 的公钥

(9) A．CA 的公钥　　B．B 的私钥　　C．A 的公钥　　D．B 的公钥

解析：数字证书是一个经证书认证中心(CA)数字签名的包含公开密钥拥有者信息以及公开密钥的文件。要验证证书的真伪，可利用 CA 的公钥验证 CA 的数字签名。数字证书采用公钥体制，即利用一对互相匹配的密钥进行加密、解密。每个用户自己设定一把特定的仅为本人所知的私有密钥(私钥)，用它进行解密和签名；同时设定一把公共密钥(公钥)并由本人公开，为一组用户所共享，用于加密和验证签名。

答案：(8) A　　(9) C

例 6　PKI 体制中，保证数字证书不被篡改的方法是__(8)__。(2013 年下半年试题 8)

(8) A．用 CA 的私钥对数字证书签名

B．用 CA 的公钥对数字证书签名

C．用证书主人的私钥对数字证书签名

D．用证书主人的公钥对数字证书签名

解析：PKI(Public Key Infrastructure)是基于公开密钥理论和技术建立起来的安全体系，如果数据在传输和处理过程中被篡改，接收方就不会收到正确的数字签名。为了保证数字证书不被篡改，采用 CA 的私钥对数字证书签名。

答案：A

例 7　在 Windows 系统中，默认权限最低的用户组是__(8)__。(2012 年上半年试题 8)

(8) A. everyone　　B. administrators　　C. power users　　D. users

解析：Administrators 组内的用户，都具备系统管理员的权限，它们拥有对这台计算机最大的控制权限，可以执行整台计算机的管理任务。Users 组员只拥有一些基本的权利，如运行应用程序，但是他们不能修改操作系统的设置、不能更改其他用户的数据、不能关闭服务器级的计算机。所有添加的本地用户账户都自动属于该组。Power Users 组内的用户具备比 Users 组更多的权利，但是比 Administrators 组拥有的权利少一些。Everyone 是个抽象组，任何一个用户都属于这个组。

答案：A

例 8　IIS 6.0 支持的身份验证安全机制有 4 种验证方法，其中安全级别最高的验证方法是__(9)__。(2012 年上半年试题 9)

(9) A. 匿名身份验证　　B. 集成 Windows 身份验证
C. 基本身份验证　　D. 摘要式身份验证

解析：匿名身份验证，不验证访问用户的身份，客户端不需要提供任何身份验证的凭据，服务端把这样的访问作为匿名的访问，并把这样的访问用户都映射到一个服务端的账户。基本身份验证，完全是把用户名和明文用明文传送到服务端验证，服务器直接验证服务器本地是否有用户跟客户端提供的用户名和密码相匹配的，如果有则通过验证。摘要式身份验证的设计目的是提高基本身份验证系统所提供的安全性，它允许服务器指定客户端应该使用的加密或散列机制，并提供了在算法中注入随机明文以协助防止重播攻击的选项。集成 Windows 身份验证可以使用 NTLM 或 Kerberos V5 身份验证，当 Internet Explorer 浏览器试图设为集成验证的 IIS 的资源时，IIS 发送两个 WWW 身份验证头，即 Negotiate 和 NTLM。基本身份验证的安全级别较低，摘要式身份验证的安全级别为中等，集成 Windows 身份验证的安全级别较高。

答案：B

例 9　通过内部发起连接与外部主机建立联系，由外部主机控制并盗取用户信息的恶意代码为__(8)__。(2011 年下半年试题 8)

(8) A. 特洛伊木马　　B. 蠕虫病毒　　C. 宏病毒　　D. CIH 病毒

解析：特洛伊木马是一种秘密潜伏的能够通过远程网络进行控制的恶意程序。控制者可以控制被秘密植入木马的计算机的一切动作和资源，是恶意攻击者进行窃取信息等的工具。特洛伊木马没有复制能力，它的特点是伪装成一个实用工具或者一个可爱的游戏，这会诱使用户将其安装在个人计算机或者服务器上。为了达到控制服务端主机的目的，木马往往要采用各种手段达到激活自己、加载运行的目的。完整的木马程序一般由两个部分组成：一个是服务端(被控制端)；一个是客户端(控制端)。“中了木马”就是指安装了木马的服务端程序，若你的计算机被安装了服务端程序，则拥有相应客户端的人就可以通过网络控制你的计算机、为所欲为，这时你计算机上的各种文件、程序以及在你计算机上使用的账号、密码就无安全性可言了。

蠕虫病毒一般是通过复制自身在互联网环境下进行传播，蠕虫病毒的传染目标是互联网内的所有计算机，局域网条件下的共享文件夹、电子邮件、网络中的恶意网页、大量存在着漏洞的服务器等都成为蠕虫传播的良好途径。

宏病毒是一种寄存在文档或模板的宏中的计算机病毒。一旦打开这样的文档，其中的

宏就会被执行，于是宏病毒就会被激活，转移到计算机上，并驻留在 Normal 模板上。从此以后，所有自动保存的文档都会"感染"上这种宏病毒，而且如果其他用户打开了感染病毒的文档，宏病毒又会转移到他的计算机上。

CIH 病毒是一种能够破坏计算机系统硬件的恶性病毒。但是 CIH 病毒只在 Windows 95、98 和 Windows Me 系统上发作，影响有限。

答案：A

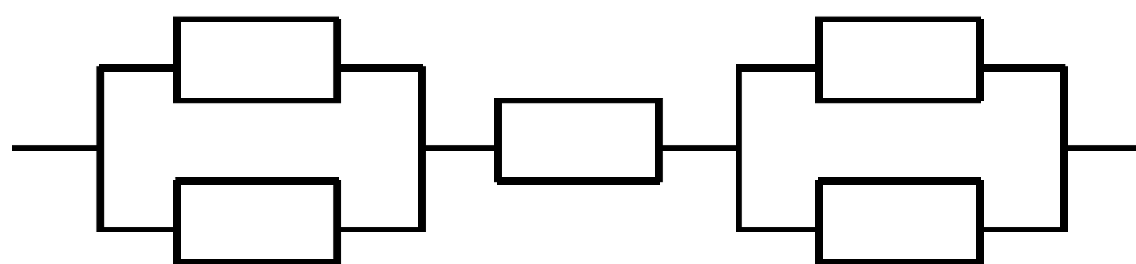
例 10　从认证中心 CA 获取用户 B 的数字证书，该证书用__(9)__作数字签名；从用户 B 的数字证书中可获得 B 的公钥。(2011 年下半年试题 9)

(9) A. CA 的公钥　　B. CA 的私钥　　C. B 的公钥　　D. B 的私钥

解析：数字证书是用电子手段来证实一个用户的身份和对网络资源的访问权限。数字证书是由权威机构(CA)采用数字签名技术，颁发给用户，用以在数字领域中证实用户其本人身份的一种数字凭证。数字签名可以用来防止电子信息被修改而作伪；或冒用别人名义发送信息；或发出后又加以否认等情况的发生。数字证书中包含认证机构的签名，该签名用的是 CA 的私钥。

答案：B

例 11　某计算机系统由下图所示部件构成，假定每个部件的千小时可靠度为 R，则该系统的千小时可靠度为__(6)__。(2011 年上半年试题 6)



(6) A. $R+2R/4$　　B. $R+R^2/4$　　C. $R(1-(1-R)^2)$　　D. $R(1-(1-R)^2)^2$

解析：两个并联部件的可靠度=$1-(1-R)(1-R)$

总可靠度=$(1-(1-R)(1-R))\times R\times(1-(1-R)(1-R))= R(1-(1-R)^2)^2$

答案：D

例 12　下面算法中，不属于公开密钥加密算法的是__(9)__。(2013 年下半年试题 9)

(9)　A. ECC　　B. DSA　　C. RSA　　D. DES

解析：ECC(Elliptic Curves Cryptography)为椭圆曲线密码编码学。RSA 是由 RSA 公司发明的一个支持变长密钥的公共密钥算法，需要加密的文件块的长度也是可变的；DSA(Digital Signature Algorithm)是数字签名算法，是一种标准的 DSS(数字签名标准)。这些都属于公开密钥加密算法。

DES 的含义是数据加密标准，其速度较快，适用于加密大量数据的场合，是一种秘密密钥加密算法。

答案：D

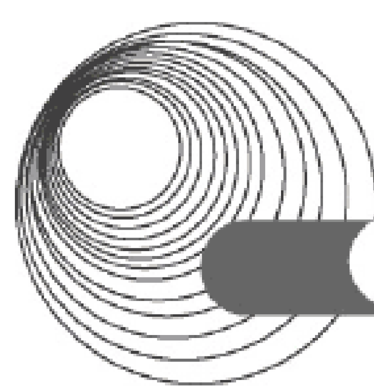
例 13　宏病毒一般感染以__(8)__为扩展名的文件。(2011 年上半年试题 8)

(8) A. EXE　　B. COM　　C. DOC　　D. DLL

解析：宏病毒是一种寄存在文档或模板的宏中的计算机病毒，它利用了在 Word 和其他办公软件中发现的特征，可感染 Word、Excel 等文件。DOC 是 Word 文档的扩展名。

答案：C

例 14　公钥体系中，私钥用于__(66)__，公钥用于__(67)__。(2010 年下半年试题 66、67)



(66) A．解密和签名 B．加密和签名 C．解密和认证 D．加密和认证

(67) A．解密和签名 B．加密和签名 C．解密和认证 D．加密和认证

解析：在公钥体系(亦即非对称密钥体制)中，每个用户都有一对密钥，即公钥和私钥，公钥对外公开，私钥由个人秘密保存。因此通常采用公钥加密，私钥解密。认证技术用于辨别用户的真伪，有基于对称加密的认证方法，也有基于公钥的认证。在基于公钥的认证中，通信双方用对方的公钥加密，用各自的私钥解密。在签名中用私钥签名消息，公钥验证签名。

答案：(66) A (67) D

例 15 杀毒软件报告发现病毒 Macro.Melissa，由该病毒名称可以推断病毒类型是 (8) ，这类病毒的主要感染目标是 (9) 。(2010 年上半年试题 8、9)

(8) A．文件型 B．引导型 C．目录型 D．宏病毒

(9) A．EXE 或 COM 可执行文件 B．Word 或 Excel 文件

C．DLL 系统文件 D．磁盘引导区

解析：Melissa 病毒是一种快速传播的能够感染那些使用 Microsoft Word 97 和 Microsoft Office 2000 的计算机宏病毒。即使不知道 Melissa 病毒是什么也没关系，因为前面有个 Macro，表明这是宏病毒。

答案：(8) D (9) B

例 16 某网站向 CA 申请了数字证书，用户通过 (9) 来验证网站的真伪。(2009 年下半年试题 9)

(9) A．CA 的签名 B．证书中的公钥

C．网站的私钥 D．用户的公钥

解析：数字证书是一个经证书认证中心(CA)数字签名的包含公开密钥拥有者信息以及公开密钥的文件。它是一段包含用户身份信息、用户公钥信息及身份验证机构数字签名的数据。身份验证机构的数字签名可以确保证书信息的真实性，用户公钥信息可以保证数字信息传输的完整性，用户的数字签名可以保证数字信息的不可否认性。

数字证书是各类终端实体和最终用户在网上进行信息交流及商务活动的身份证明，在电子交易的各个环节，交易的各方都需验证对方数字证书的有效性，从而解决相互间的信任问题。

对于数字信息的安全需求，通过以下手段加以解决：数据保密性——加密，数据的完整性和不可否认性——数字签名，身份鉴别——数字证书与数字签名。

目前采用数字证书技术来实现网上信息传输双方的身份验证和信息传输安全，从而实现对传输信息的机密性、真实性、完整性和不可否认性。

答案：A

例 17 安全需求可划分为物理线路安全、网络安全、系统安全和应用安全。下面的安全需求中属于系统安全的是 (8) ，属于应用安全的是 (9) 。(2015 年上半年试题 8、9)

(8) A．机房安全 B．入侵检测

C．漏洞补丁管理 D．数据库安全

(9) A．机房安全 B．入侵检测

C．漏洞补丁管理 D．数据库安全

解析：物理线路安全包括通信线路的安全、物理设备的安全、机房的安全等。网络安全包括网络层身份认证、网络资源的访问控制、数据传输的保密与完整性、远程接入的安全、域名系统的安全、路由系统的安全、入侵检测的手段、网络设施防病毒等。系统安全问题来自网络内使用的操作系统的安全。主要表现在 3 个方面：一是操作系统本身的缺陷带来的不安全因素，主要包括身份认证、访问控制、系统漏洞等；二是对操作系统的安全配置问题；三是病毒对操作系统的威胁。应用安全问题主要由提供服务所采用的应用软件和数据的安全性产生，包括 Web 服务、电子邮件系统、DNS 等。

答案：(8)C　(9)D

1.3.3　同步练习

1.　若某计算机系统是由 500 个元器件构成的串联系统，且每个元器件的失效率约为 10^{-7}/h，在不考虑其他因素对可靠性的影响时，该计算机系统的平均故障间隔时间为____h。

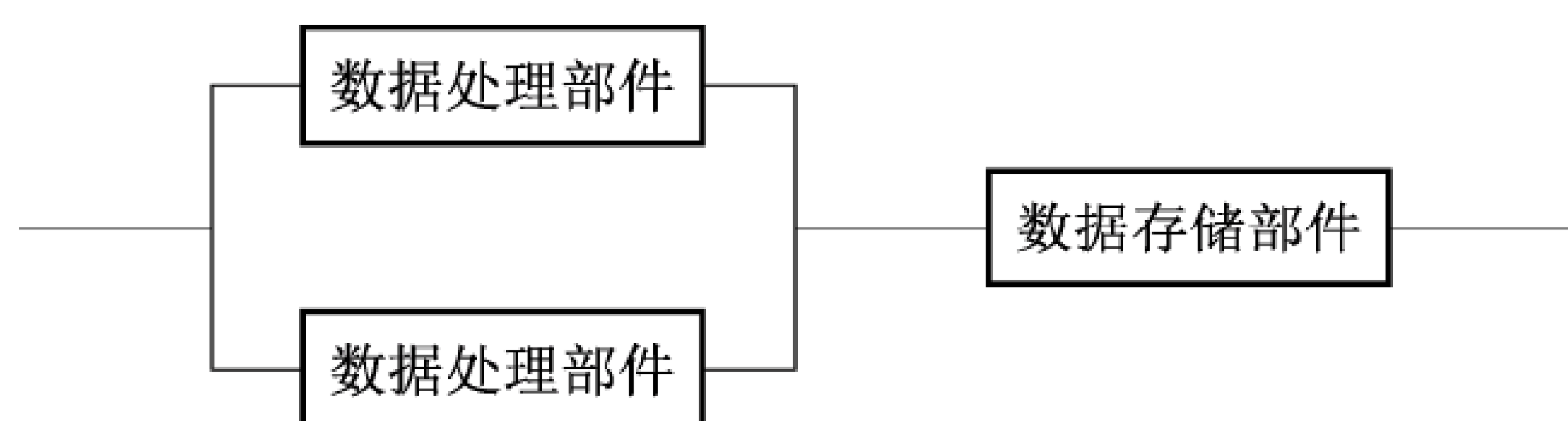
A．2×10^4　　B．5×10^4　　C．2×10^5　　D．5×10^5

2.　相对于 DES 算法而言，RSA 算法的 (1) ，因此，RSA (2) 。

(1)　A．加密密钥和解密密钥是不相同的
B．加密密钥和解密密钥是相同的
C．加密速度比 DES 要高
D．解密速度比 DES 要高

(2)　A．更适用于对文件加密
B．保密性不如 DES
C．可用于对不同长度的消息生成消息摘要
D．可以用于数字签名

3．某数据处理软件包括两个完全相同的数据处理部件和一个数据存储部件，且采用下图所示的容错方案。当数据处理部件的可靠性为 0.6 时，为使整个软件系统的可靠性不小于 0.66，则数据存储部件的可靠性至少应为_____。

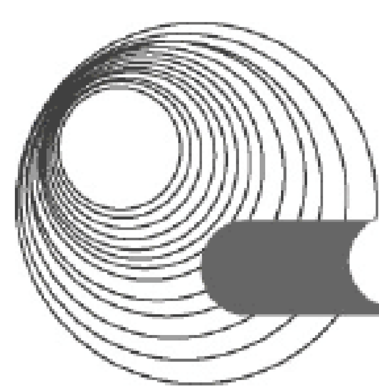


容错方案

A．0.6　　B．0.66　　C．0.79　　D．1.0

4.　系统响应时间和作业吞吐量是衡量计算机系统性能的重要指标。对于一个持续处理业务的系统而言，_____，表明其性能越好。

A．响应时间越短，作业吞吐量越小
B．响应时间越短，作业吞吐量越大
C．响应时间越长，作业吞吐量越大
D．响应时间不会影响作业吞吐量

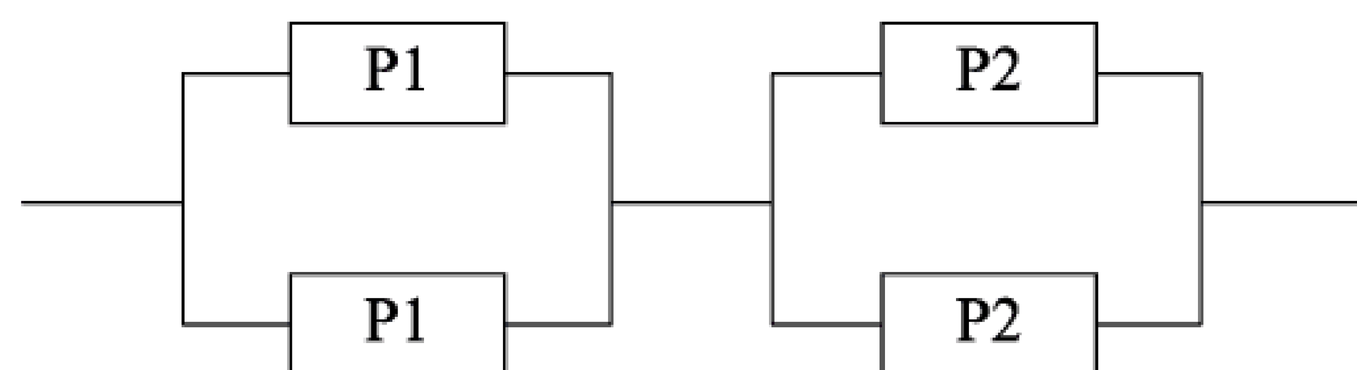


5．某 Web 网站向 CA 申请了数字证书。用户登录该网站时，通过验证(1)，可确认该数字证书的有效性，从而__(2)__。

(1) A．CA 的签名　　B．网站的签名
　　C．会话密钥　　D．DES 密码

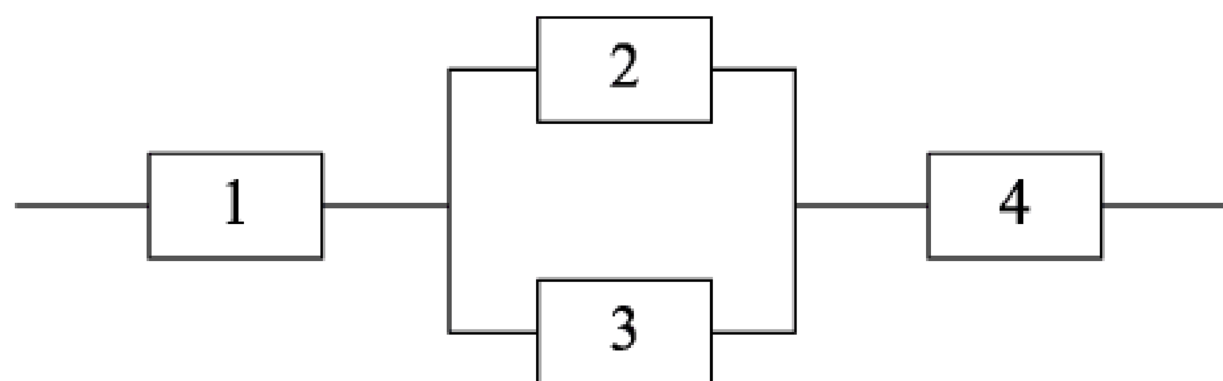
(2) A．向网站确认自己的身份　　B．获取访问网站的权限
　　C．和网站进行双向认证　　D．验证该网站的真伪

6．某大型软件系统按功能可划分为两段，即 P1 和 P2。为提高系统可靠性，软件应用单位设计了如下图所示的软件冗余容错结构，其中 P1 和 P2 均有一个与其完全相同的冗余备份。若 P1 的可靠度为 0.9，P2 的可靠度为 0.9，则整个系统的可靠度是_____。



A．0.6561　　B．0.81　　C．0.9801　　D．0.9

7．某系统的可靠性结构框图如下图所示。该系统由 4 个部件组成，其中 2、3 两部件并联冗余，再与 1、4 两部件串联构成。假设部件 1、2、3 的可靠度分别为 0.90、0.70、0.70。若要求该系统的可靠度不低于 0.75，则进行系统设计时，分配给部件 4 的可靠度至少应为_____。



A．$\frac{0.75}{0.9\times(1-0.7)^2}$　　B．$\frac{0.75}{0.9\times[(1-(1-0.7)^2)]}$

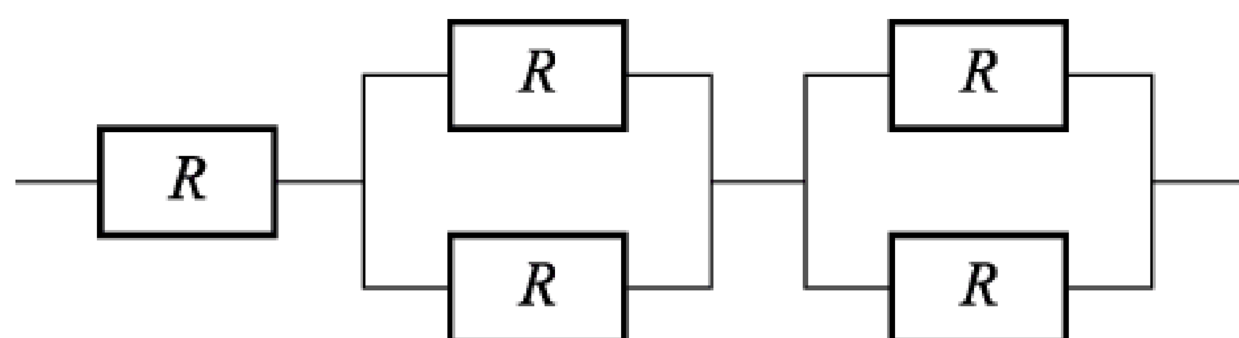
C．$\frac{0.75}{0.9\times(1-0.7\times0.7)^2}$　　D．$\frac{0.75}{0.9\times(0.7+0.7)}$

8．系统响应时间和作业吞吐量是衡量计算机系统性能的重要指标。对于一个持续处理业务的系统而言，其_____。

A．响应时间越短，作业吞吐量越小
B．响应时间越短，作业吞吐量越大
C．响应时间越长，作业吞吐量越大
D．响应时间不会影响作业吞吐量

9．某计算机系统由下图所示的部件构成，假定每个部件的千小时可靠度 R 均为 0.9，则该系统的千小时可靠度约为_____。

A．0.882　　B．0.951　　C．0.9　　D．0.99



10．某计算机的时钟频率为 400MHz，测试该计算机的程序使用 4 种类型的指令。每种指令的数量及所需指令时钟数(CPI)如下表所示，则该计算机的指令平均时钟数为(1)，该计算机的运算速度约为 (2) MIPS。

指令类型	指令数目/条	每条指令需时钟数
1	160000	1
2	30000	2
3	24000	4
4	16000	8

(1) A．1.85　　B．1.93　　C．2.36　　D．3.75

(2) A．106.7　　B．169.5　　C．207.3　　D．216.2

11．“冲击波”病毒属于 (1) 类型的病毒，它利用 Windows 操作系统的 (2) 漏洞进行快速传播。

(1) A．蠕虫　　B．文件
　　C．引导区　　D．邮件

(2) A．CGI 脚本　　B．RPC
　　C．DNS　　D．IMAP

1.3.4　同步练习参考答案

1．A　　2．(1) A　(2) D　　3．C　　4．B

5．(1) A　(2) D　　6．C　　7．C　　8．B

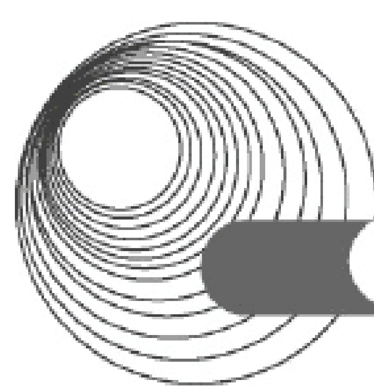
9．A　　10．(1) B　(2) C　　11．(1) A　(2) B

1.4　本 章 小 结

本章知识点在 2013 年的新大纲有一定改动，在新大纲中添加了数据表示和校验码的相关知识，去掉了数学基础知识和计算机故障的诊断与容错知识点。另外，还有一些描述方面的调整。

本章主要要求考生掌握计算机系统的硬件组成、计算机工作的基本原理、计算机的体系结构、存储系统以及可靠性与系统性检测。主要掌握计算机的一些预备知识、系统分类；掌握各种数制及其相互转换，CPU 的基本结构，存储系统的基本原理，包括内存、外存、高速缓存等的基本概念；熟悉 I/O 设备的基本知识；RISC 和 CISC 的基本概念及其区别；理解流水线的基本概念；了解阵列处理机和并行处理机，要求会对系统的可靠性进行分析。

分析历年的考题可以看出，在历年的考试中这个章节的内容都要考 7～9 道题目，每年的考点都不同，并且试题难度从总体来看比较大。



这个章节的内容相对较多，概念杂，知识点散，有的需要记忆，有的需要计算，对考生的综合能力要求较高。只有把每个知识点都搞清楚，同时多做练习，才能达到好的学习效果。

1.5 达标训练题及参考答案

1.5.1 达标训练题

1. 两个同符号的数相加或异符号的数相减，所得结果的符号位 SF 和进位标志 CF 进行_____运算结果为 1 时，表示运算的结果产生溢出。

A．与　　B．或　　C．与非　　D．异或

2. 若浮点数的阶码用移码表示，尾数用补码表示。两规格化浮点数相乘，最后对结果规格化时，右规的右移位数最多为_____位。

A．1　　B．2　　C．尾数位数　　D．尾数位数-1

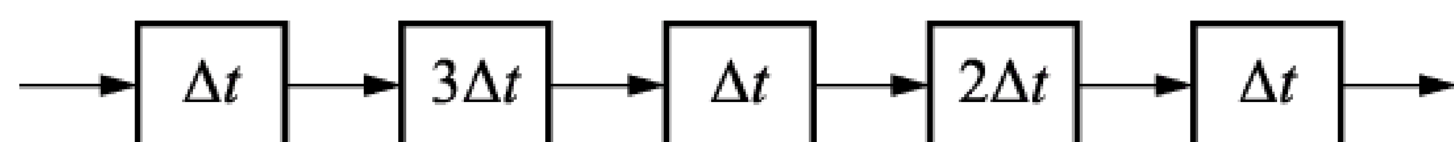
3. 高速缓存(Cache)与主存间采用全相联地址映像方式，高速缓存的容量为 4MB，分为 4 块，每块 1MB，主存容量为 256MB。若主存读写时间为 30ns，高速缓存的读写时间为 3ns，平均读写时间为 3.27ns，则该高速缓存的命中率为 (1) %。若地址变换表如下表所示，则主存地址为 8888888H 时，高速缓存地址为 (2) H。

0	38H
1	88H
2	59H
3	67H

(1) A．90　　B．95　　C．97　　D．99

(2) A．488888　　B．388888　　C．288888　　D．188888

4. 某指令流水线由 5 段组成，各段所需要的时间如下图所示。连续输入 10 条指令时的吞吐率为_____。



A．10Δ*t*/70　　B．10Δ*t*/49　　C．10Δ*t*/35　　D．10Δ*t*/30

5. 驻留在多个网络设备上的程序在短时间内同时产生大量的请求消息冲击某 Web 服务器，导致该服务器不堪重负，无法正常响应其他合法用户的请求，这属于_____。

A．网上冲浪　　B．中间人攻击　　C．DDoS 攻击　　D．MAC 攻击

1.5.2 参考答案

1．D　　2．A　　3．(1) D　(2) D

4．C　　5．C

第 2 章　程序设计语言基础

大纲要求：

- 汇编、编译、解释系统的基础知识和基本工作原理。
- 程序设计语言的基本成分——数据、运算、控制和传输以及过程(函数)调用。
- 各类程序设计语言的主要特点和适用情况。

2.1　程序设计基础知识

2.1.1　考点辅导

2.1.1.1　程序设计语言的基本概念

1. 低级语言和高级语言

1)　低级语言

通常称机器语言和汇编语言为低级语言。机器语言是指用 0、1 字符串组成的机器指令序列，是最基本的计算机语言；汇编语言是指用符号表示指令的语言。

2)　高级语言

高级语言是从人类的逻辑思维角度出发、面向各类应用的程序语言，抽象程度大大提高，需要经过编译成特定机器上的目标代码才能执行。这类语言与人们使用的自然语言比较接近，大大提高了程序设计的效率。

2. 编译程序和解释程序

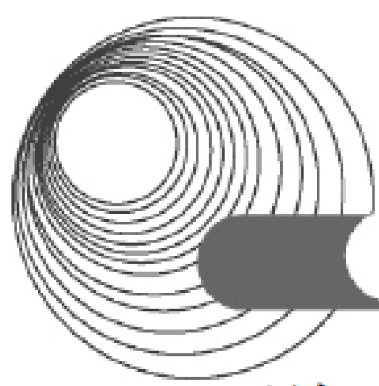
尽管人们可以借助高级语言和计算机进行交互，但是计算机仍然只能理解和执行由 0、1 序列构成的机器语言，因此高级程序语言需要翻译，担任这一任务的程序称为语言处理程序。用某种高级语言或汇编语言编写的程序称为源程序，源程序不能直接在计算机上执行。如果源程序是使用汇编语言编写的，则需要一个称为汇编程序的翻译程序将其翻译成目标程序后才能执行。如果源程序是使用某种高级语言编写的，则需要相应的解释程序或编译程序对其进行翻译，然后才能在机器上执行。

解释程序也称为解释器，它或者直接解释执行源程序，或者将源程序翻译成某种中间表示形式后再执行；而编译程序(编译器)则是将源程序翻译成目标语言程序，然后在计算机上执行目标程序。

3. 程序设计语言的定义

下面介绍关于程序设计语言的定义。

(1)　语法。由程序设计语言的基本符号组成程序中的各个语法成分(包括程序)的一组规



则，其中由基本符号构成的符号(单词)书写规则称为词法规则，由符号(单词)构成语法成分的规则称为语法规则。程序语言的语法可通过形式语言进行描述。

(2) 语义。程序语言中按语法规则构成的各个语法成分的含义，可分为静态语义和动态语义。

(3) 语用。表示构成语言的各个记号和使用者的关系，涉及符号的来源、使用和影响。

(4) 语境。理解和实现程序设计语言的环境，包括编译环境和运行环境。

4. 程序设计语言的分类

1) 命令式程序设计语言

命令式程序设计语言是基于动作的语言，在这种语言中，计算被看作动作的序列。命令式语言族开始于 FORTRAN、PASCAL 和 C 语言，体现了命令式程序设计的关键思想。

2) 面向对象的程序设计语言

(1) 对象。对象是指人们要进行研究的任何事物，它具有状态和操作。面向对象的语言把状态和操作封装于对象实体之中，并提供一种访问机制。用户只能通过向允许公开的操作提出要求，才能查询和修改对象的状态。

(2) 类。类是面向对象语言必须提供的、由用户定义的数据类型，它将具有相同状态、操作和访问机制的多个对象抽象成一个对象类。在定义类以后，属于这种类的一个对象被称为类实例或类对象。

(3) 继承。继承是面向对象语言的另一个基本要素。继承实现了一般与特殊的关系，解决了软件的可重用性和可扩充性的问题。

3) 函数式程序设计语言

函数式程序设计语言是一类以λ-演算为基础的语言。该语言的代表是 LISP 语言，其中大量使用了递归。

函数是一种对应规则(映射)，它是定义域中的每个元素和值域中唯一的元素相对应。函数可以看成一种程序，其输入就是定义在左边括号中的变量，可以将输入组合起来产生一个规则，组合过程中也可以使用其他函数或函数本身。这种用函数和表达式建立程序的方法就是函数式程序设计。函数型程序设计语言的优点之一就是表达式中出现的任何函数都可以用其他函数来代替，只要这些函数调用产生相同的值。

4) 逻辑型程序设计语言

逻辑型程序设计语言是一类以形式逻辑为基础的语言。该语言的代表是建立在关系理论和一阶谓词理论基础上的 Prolog 语言。Prolog 语言具有很强的推理功能，适用于书写自动定理证明、专家系统以及自然语言理解等问题的程序。

2.1.1.2 程序设计语言的基本成分

1. 数据成分

程序语言的数据成分是指一种程序语言的数据类型。

1) 常量和变量

按照程序运行时数据的值能否改变，将数据分为常量和变量。程序中的数据对象可以具有左值和(或)右值，左值是指存储单元(或地址、容器)，右值是指具体值(或内容)。变量具有左值和右值，在程序运行过程中其右值可以改变；常量只有右值，在程序运行过程中

其右值不能改变。

2)　全局量和局部量

按数据的作用域范围，数据可分为全局量和局部量。系统为全局变量分配的存储空间在程序运行的过程中一般是不改变的，而为局部变量分配的存储单元是动态改变的。

3)　数据类型

按照数据组织形式的不同可将数据分为基本类型、用户定义类型、构造类型及其他类型。C(C++)的数据类型如下。

- 基本类型：整型(int)、字符型(char)、实型(float、double)和布尔类型(bool)。
- 特殊类型：空类型(void)。
- 用户定义类型：枚举类型(enum)。
- 构造类型：数组、结构体和共用体。
- 指针类型：type *。
- 抽象数据类型：类类型。

其中，布尔类型和类类型是 C++在 C 语言的基础上扩充的。

2. 运算成分

程序语言的运算成分是指允许使用的运算符号及运算规则。大多数高级程序语言的基本运算可以分成算术运算、关系运算和逻辑运算，有些语言还提供位运算。运算符号的使用与数据类型密切相关。为了确保运算结果的唯一性，运算符号要规定优先级和结合性，必要时还要使用圆括号。

3. 控制成分

控制成分指明语言允许表述的控制结构，程序员使用控制成分来构造程序中的控制逻辑。

1)　顺序结构

在顺序结构中，计算过程从所描述的第一个操作开始，按顺序依次执行后续的操作，直到执行完序列的最后一个操作。顺序结构内也可以包含其他控制结构。

2)　选择结构

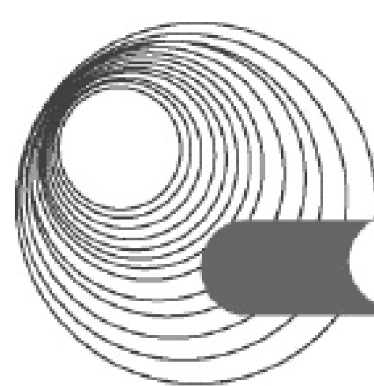
选择结构提供了在两种或多种分支中选择执行其中一个分支的逻辑。基本的选择结构是指定一个条件 P，然后根据条件的成立与否决定控制流走计算 A 还是走计算 B，从两个分支中选择一个执行。选择结构中的计算 A 或计算 B 还可以包含顺序、选择和重复结构。程序语言中通常还提供简化了的选择结构，也就是没有计算 B 的分支结构。

3)　循环结构

循环结构描述了重复计算的过程，通常包括 3 个部分，即初始化、需要重复计算的部分和重复的条件。其中初始化部分有时在控制的逻辑结构中不进行显式表示。循环结构主要有两种形式，即 while 型重复结构和 do-while 型重复结构。

4)　C(C++)语言提供的控制语句

(1) 复合语句。复合语句用于描述顺序控制结构。复合语句是一系列用“{”和“}”括起来的声明和语句，其主要作用是将多条语句组成一个可执行单元。复合语句是一个整体，要么全部执行，要么一条语句也不执行。



(2) if 语句和 switch 语句。这两种语句用于实现选择结构。

① if 语句实现的是双分支的选择结构，其一般形式如下：

```
if(表达式)语句 1;else 语句 2;
```

其中，语句 1 和语句 2 可以是任何合法的 C(C++)语句，当语句 2 为空语句时，可以简化为

```
if(表达式) 语句;
```

使用 if 语句时，需要注意的是 if 和 else 的匹配关系。C 语言规定，else 总是与离它最近的尚没有 else 与其匹配的 if 相匹配。

② switch 语句描述了多分支的选择结构，其一般形式如下：

```
switch(表达式){
case 常量表达式 1: 语句 1;
case 常量表达式 2: 语句 2;
...
case 常量表达式 n: 语句 n;
default: 语句 n+1;
}
```

执行 switch 语句时，首先计算表达式的值，然后用所得的值与列举的常量表达式值依次比较，若任一常量表达式都不能与所得的值相匹配，则执行 default 的“语句序列 n+1”，然后结束 switch 语句。

表达式可以是任何类型，常用的是字符型或整型表达式。多个常量表达式可以共用一个语句组。语句组可以包括任何可执行语句，且无须用“{”和“}”括起来。

(3) 循环语句。C(C++)语言提供了 3 种形式的循环语句，用于描述循环计算的控制结构。

① while 语句。while 语句描述了先判断条件再执行循环体的控制结构，其一般形式如下：

```
while(条件表达式) 循环体语句;
```

② do-while 语句。do-while 语句描述了先执行循环体再判断条件的控制结构，其一般格式如下：

```
do
    循环体语句;
while(条件表达式);
```

③ for 语句。for 语句的基本格式如下：

```
for(表达式 1;表达式 2;表达式 3) 循环体语句;
```

可用 while 语句等价地表示为

```
表达式 1;
while(表达式 2){
    循环体语句;
    表达式 3;
}
```

4. 函数

函数是程序模块的主要成分，它是一段具有独立功能的程序。函数的使用涉及 3 个概念，即函数定义、函数声明和函数调用。

(1) 函数定义。包括函数首部和函数体两个部分。函数的定义描述了函数做什么和怎么做。

(2) 函数声明。函数应该先声明后引用。函数声明定义了函数原型。声明函数原型的目的在于告诉编译器传递给函数的参数个数、类型以及函数返回值的类型，参数表中仅需要依次列出函数定义中的参数类型。函数原型可以使编译器检查源程序中对函数的调用是否正确。

(3) 函数调用。当需要在一个函数(称为主调函数)中使用另一个函数(称为被调函数)实现的功能时，便以函数名字进行调用，称为函数调用。调用函数和被调用函数之间交换信息的方法主要有两种：一种是由被调用函数把返回值返回给主调函数；另一种是通过参数带回信息。函数调用时实参和形参间交换信息的方法有传值调用和引用调用两种。

① 传值调用(Call by Value)。若实现函数调用时实参向形式参数传递相应类型的值(副本)，则称为传值调用。这种方式下形式参数不能向实际参数传递信息。在 C 语言中，要实现被调用函数对实际参数的修改，必须用指针作形参。即调用时需要先对实参进行取地址运算，然后将实参的地址传递给指针形参，本质上仍属于传值调用。这种方式实现了间接内存访问。

② 引用调用(Call by Reference)。引用是 C++中增加的数据类型，当形式参数为引用类型时，形参名实际上是实参的别名，函数中对形参的访问和修改实际上就是针对相应实际参数所作的访问和改变。

2.1.2 典型例题分析

例 1 以下关于解释程序和编译程序的叙述中，正确的是 (20) 。(2013 年上半年试题 20)

(20) A. 编译程序和解释程序都生成源程序的目标程序
B. 编译程序和解释程序都不生成源程序的目标程序
C. 编译程序生成源程序的目标程序，解释程序则不然
D. 编译程序不生成源程序的目标程序，而解释程序反之

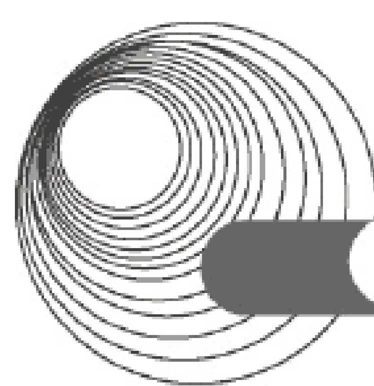
解析：编译程序也称编译器，将源程序翻译成目标语言程序，然后在计算机上运行目标程序。虽然执行效率高，但编写出来的程序可读性很差，且难以修改和维护。

解释程序：也称解释器，它或者直接解释执行源程序，或者将源程序翻译成某种中间代码后再加以执行。

答案：C

例 2 以下关于传值调用与引用调用的叙述中，正确的是 (21) 。(2013 年上半年试题 21)

① 在传值调用方式下，可以实现形参和实参间双向传递数据的效果
② 在传值调用方式下，实参可以是变量，也可以是常量和表达式
③ 在引用调用方式下，可以实现形参和实参间双向传递数据的效果



④ 在引用调用方式下，实参可以是变量，也可以是常量和表达式

(21) A. ①③　　B. ①④　　C. ②③　　D. ②④

解析：传值调用最显著的特征是被调用的函数内部对形参的修改不影响实参的值。传值调用的参数传递和被调用函数内对参数的使用主要按下述原则：函数定义时形参被当作局部变量看待，在函数被调用时为形参分配存储单元；调用函数前，首先计算实参的值，调用时将实参的值放入形参的存储单元；被调用函数内部对形参单元中的数据进行直接访问。

引用调用是将实参的地址传递给形参，使得形参的地址就是对应实参的地址。引用调用的参数传递和被调用函数内对参数的使用主要按下述原则处理：函数定义时形参被当作局部变量看待，在函数被调用时为形参分配存储单元；调用时将实参的地址放入形参的存储单元；被调用函数内部对形参单元中的数据(地址)进行间接访问。

答案：C

例 3　可用于编写独立程序和快速脚本的语言是__(20)__。(2012 年下半年试题 20)

(20) A. Python　　B. Prolog　　C. Java　　D. C#

解析：Python 是一种面向对象的解释型程序设计语言，可用于编写独立程序、快速脚本和复杂应用的原型。Python 也是一种脚本语言，它支持对操作系统底层的访问。

Prolog 是一种逻辑型语言。Prolog 程序是一系列事实、数据对象或事实间的具体关系和规则的集合。Prolog 有很强的推理功能，适用于书写自动定理证明、专家系统、自然语言理解等问题的程序。

Java 是一种面向对象的程序设计语言，能开发应用在 Internet 上且具有软、硬件独立性和交互能力的程序。Java 语言的程序可以一次编写而到处运行。

C#是微软公司发布的一种面向对象的、运行于.NET Framework 之上的高级程序设计语言。C#看起来与 Java 有着惊人的相似：它包括了如单一继承、接口、与 Java 几乎同样的语法和编译成中间代码再运行的过程。但是 C#与 Java 有着明显的不同，它借鉴了 Delphi 的一个特点，与 COM(组件对象模型)是直接集成的，而且它是微软公司.NET Windows 网络框架的主角。

答案：A

例 4　将高级语言源程序翻译成目标程序的是__(48)__。(2012 年下半年试题 48)

(48) A. 解释程序　　B. 编译程序　　C. 链接程序　　D. 汇编程序

解析：编译程序的功能是把某高级语言书写的源程序翻译成与之等价的目标程序。解释程序是另一种语言处理程序，在词法、语法和语义分析方面与编译程序的工作原理基本相同，但在运行用户程序时，它直接执行源程序或源程序的中间表示形式。解释程序不产生源程序的目标程序，这是它和编译程序的主要区别。

答案：B

例 5　以下关于程序设计语言的叙述中，错误的是__(20)__。(2015 年上半年试题 20)

(20) A. 程序设计语言的基本成分包括数据、运算、控制和传输等
B. 高级程序设计语言不依赖于具体的机器硬件
C. 程序中局部变量的值在运行时不能改变
D. 程序中常量的值在运行时不能改变

解析：变量具有左值和右值，在程序运行过程中，局部变量的右值可以改变。

答案：C

例 6　函数(过程)调用时，常采用传值与传地址两种方式在实参和形参间传递信息。以下叙述中，正确的是__(50)__。(2012 年上半年试题 50)

(50) A. 在传值方式下，将形参的值传给实参，因此，形参必须是常量或变量

B. 在传值方式下，将实参的值传给形参，因此，实参必须是常量或变量

C. 在传地址方式下，将形参的值传给实参，因此，形参必须有地址

D. 在传地址方式下，将实参的值传给形参，因此，实参必须有地址

解析：形式参数就是过程定义中函数名后括号中所带的参数；实际参数是在调用点表示向被调用过程传递的数据。在函数调用时，数据传递的方向是从实参到形参。只是采用传值传递方式时，传递的是数值，这个数值只要是确定的即可，可以是常理、变量或表达式等。而采用传址传递方式时，传递的是地址，因此实参必须有地址。

答案：D

例 7　编译器和解释器是两种基本的高级语言处理程序。编译器对高级语言源程序的处理过程可以划分为词法分析、语法分析、语义分析、中间代码生成、代码优化、目标代码生成等阶段，其中，__(20)__并不是每个编译器都必需的，与编译器相比，解释器__(21)__。(2015 年下半年试题 20、21)

(20) A．词法分析和语法分析　　B．语义分析和中间代码生成

C．中间代码生成和代码优化　　D．代码优化和目标代码生成

(21) A．不参与运行控制，程序执行的速度慢

B．参与运行控制，程序执行的速度慢

C．参与运行控制，程序执行的速度快

D．不参与运行控制，程序执行的速度快

解析：在编译过程中中间代码的生成与优化不是必需的，但用中间代码有很多的好处，最重要的是两点：①便于实现优化，使最终代码的质量更高；②通过中间代码实现前后级分离，在多系统、多语言开发时，可大幅提高整体开发效率，减少开发成本、缩短开发周期。所以实际的编译系统多数都会使用中间代码。

在解释器上运行程序比直接运行编译过的代码要慢，是因为解释器每次都必须去分析并转译它所运行到的程序行，而编译过的程序直接运行即可。

答案：(20) C　(21) B

例 8　若 C 程序的表达式中引用了未赋初值的变量，则__(20)__。(2011 年下半年试题 20)

(20) A. 编译时一定会报告错误信息，该程序不能允许

B. 可以通过编译并运行，但运行时一定会报告异常

C. 可以通过编译，但链接时一定会报告错误而不能运行

D. 可以通过编译并运行，但运行结果不一定是期望的结果

解析：全局变量和静态局部变量在定义时如果没有初始化，编译时会自动初始化为 0；而普通的局部变量，如果没有初始化，则其值是一个随机数。在 C 程序表达式中，只要引用的变量定义了，就可以通过编译并运行，但运行结果不一定是期望的结果。

答案：D

例 9 函数 t()、f()的定义如下所示，若调用函数 t 时传递给 x 的值为 5，并且调用函数 f()时，第一个参数采用传值(call by value)方式，第二个参数采用引用(call by reference)方式，则函数 t 的返回值为__(50)__。(2015 年下半年试题 50)

```
t(int x)             f(int r, int &s)
int a;               int x;
a=3*x+1;             x=2*s+1; s=x+r;
f(x,a);              r=x-1;
return a-x;          return;
```

(50) A．33　　B．22　　C．11　　D．负数

解析： 在函数 t 中，执行语句 a=3*x+1，得 a=16；调用 f(x,a)时，将 x 的值 5、a 的值 16 传递给函数 f 的形参 r 和 s。由于 r 采用的是传值方式，函数调用后不会改变 x 的值；而参数 s 采用的是引用方式，函数调用后 a 的值发生改变。函数 f 执行完成后 a 的值变为 38，x 的值不变，为 5，因此函数 t 的返回值为 a−x= 38−5=33。

答案： A

例10 在引用调用方式下进行函数调用是将__(21)__。(2014年上半年试题21)

(21) A．实参的值传递给形参　　B．实参的地址传递给形参

C．形参的值传递给实参　　D．形参的地址传递给实参

解析： 引用调用是把实参(如int a)的地址(&a)赋给形参(指针变量，比如*b,这时b=&a,即b指向变量a),如果*b(也即a对应的内存空间)发生变化，也就是变量a的值发生了变化。

答案： B

例 11 以下关于变量和常量的叙述中，错误的是__(20)__。(2010 年下半年试题 20)

(20) A．变量的取值在程序运行过程中可以改变，常量则不行

B．变量具有类型属性，常量则没有

C．变量具有对应的存储单元，常量则没有

D．可以对变量赋值，不能对常量赋值

解析： 常量是在程序运行过程中值不可以改变的数据。根据数据的组织类型的不同，可以将数据分为基本数据类型、用户自定义数据类型、构造类型等。变量具有类型属性，常量也有数据类型，如整数常量、字符串常量等。

答案： B

例 12 下面 C 程序段中 count++语句执行的次数为__(64)__。(2010 年下半年试题 64)

```
for(int i=1;i<=11;i*=2)
    for(int j=1;j<=i;j++)
        count++;
```

(64) A．15　　B．16　　C．31　　D．32

解析： 第 1 轮循环，i=1，count++执行 1 次，然后 i=2；第 2 轮循环，i=2，count++执行 2 次，然后 i=4；第 3 轮循环，i=4，count++执行 4 次，然后 i=8；第 4 轮循环，i=8，count++执行 8 次，然后 i=16，i>11，不满足循环条件，循环结束。可以计算 count++语句执行的次数为 1+2+4+8=15。

答案： A

例 13 程序的 3 种基本控制结构是__(33)__。(2010 年上半年试题 33)

(33) A．过程、子程序和程序　　B．顺序、选择和重复
C．递归、堆栈和队列　　D．调用、返回和跳转

解析：程序的 3 种基本控制结构是顺序结构、选择结构和重复结构。

答案：B

例 14　函数调用时，基本的参数传递方式有传值与传址两种，(20)。(2009 年上半年试题 20)

(20) A．在传值方式下，形参将值传给实参
B．在传值方式下，实参不能是数组元素
C．在传址方式下，形参和实参间可以实现数据的双向传递
D．在传址方式下，实参可以是任意的变量和表达式

解析：首先看 A 选项。在传值方式下，对应的实参和形参是两个独立的实体，占用不同的内存单元，调用函数时，系统把实参值复制一份给形参，便断开两者的联系，形参值的改变对实参无影响。因此，“传值”是单向的，只能由实参传递给形参。

B 选项，形参为传值方式下的简单变量，实参可以是与其同类型的常量、变量、数组元素或表达式。

C 选项，在传址方式下，函数调用时，系统将实参的地址传递给形参，即这时参数传递的不是数据本身，而是数据在内存中的地址。所以在被调用函数中，任何对形式参数的访问，都被认为是对形式参数的间接访问。实参与形参占用不同的存储单元，传递方式是双向的，形参值的改变将影响实参值。故 C 选项正确。

D 选项，形参为传址方式时，实参如果为常量或表达式，则传址无效，相当于传值方式。

答案：C

例 15　下面关于程序语言的叙述，错误的是(22)。(2009 年上半年试题 22)

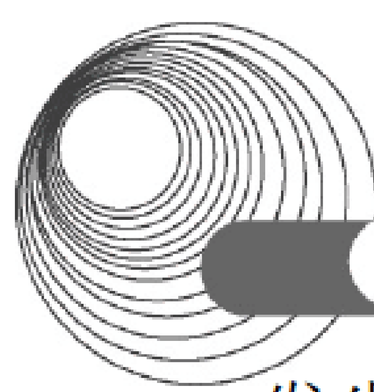
(22) A．脚本语言属于动态语言，其程序结构可以在运行中改变
B．脚本语言一般通过脚本引擎解释执行，不产生独立保存的目标程序
C．PHP、JavaScript 属于静态语言，其所有成分可在编译时确定
D．C#、Java 语言属于静态语言，其所有成分可在编译时确定

解析：脚本是一种特定的描述性语言，是依据一定的格式编写的可执行文件，又称作宏或批处理文件。脚本通常可以由应用程序临时调用并执行。脚本语言一般都是以文本形式存在，类似于一种命令。有些程序，如 C、C++、Java 等则必须先经过编译，将源代码转换为二进制代码之后才可执行。而像 Perl、JavaScript、VBScript 等则不需要事先编译，只要利用合适的解释器便可以执行代码。

动态类型语言是指在运行期间才去做数据类型检查的语言。也就是说，在用动态类型的语言编程时，永远也不用给任何变量指定数据类型，该语言会在你第一次赋值给变量时，在内部将数据类型记录下来，不用编译即可运行。Python 和 Ruby 就是一种典型的动态类型语言，其他的各种脚本语言如 JavaScript 属于动态类型语言。静态类型语言的数据类型是在编译期间检查的，也就是说，在写程序时要声明所有变量的数据类型。C/C++是静态类型语言的典型代表，其他的静态类型语言还有 C#、Java 等。故错误的为选项 C。

答案：C

例16　在某C/C++程序中，整型变量a的值为0且应用在表达式“c=b/a”中，则最可能



发生的情形是 (50) 。(2014年上半年试题50)

(50) A．编译时报告有语法错误　　B．编译时报告有逻辑错误
C．运行时报告有语法错误　　D．运行时产生异常

解析：编译时a的值无法确定，表达式“c=b/a”符合C/C++语言的语法逻辑，编译时不会报错。运行时，代入a的值，发生错误。

答案：D

2.1.3 同步练习

1. 程序设计语言一般都提供多种循环语句，如实现先判断循环条件再执行循环体的while 语句和先执行循环体再判断循环条件的 do-while 语句。关于这两种循环语句，在不改变循环体的条件下，_____是正确的。

A．while 语句的功能可由 do-while 语句实现
B．do-while 语句的功能可由 while 语句实现
C．若已知循环体的次数，则只能使用 while 语句
D．循环条件相同时，do-while 语句的执行效率更高

2. 下列叙述中错误的是_____。

A．面向对象程序设计语言可支持过程化的程序设计
B．给定算法的时间复杂性与实现该算法所采用的程序设计语言无关
C．与汇编语言相比，采用脚本语言编程可获得更高的运行效率
D．面向对象程序设计语言不支持对一个对象的成员变量进行直接访问

2.1.4 同步练习参考答案

1. C
2. C

2.2 语言处理程序基础

2.2.1 考点辅导

2.2.1.1 汇编语言的基本原理

1. 汇编语言

汇编语言是为特定的计算机或计算机系统设计的面向机器的符号化的程序设计语言。用汇编语言编写的程序称为汇编语言源程序。

汇编语言源程序由若干条语句组成。一个程序中可以有 3 类语句，即指令语句、伪指令语句和宏指令语句。

(1) 指令语句。又称为机器指令语句，汇编后能产生相应的机器代码，被 CPU 直接识别并执行相应的操作。指令语句可分为传送指令、算术运算指令、逻辑运算指令、移位指令、转移指令和处理机控制指令等。

(2) 伪指令语句。指示汇编程序在对源程序进行汇编时完成某些工作。与指令语句的区别是：伪指令语句经汇编后不产生机器代码，另外，伪指令语句所指示的操作是在源程序被汇编时完成的，而指令语句的操作必须在程序运行时完成。伪指令语句包括常数定义伪指令语句、存储定义伪指令语句、开始伪指令语句和结束伪指令语句。

(3) 宏指令语句。将多次重复使用的程序段定义为宏。宏的定义必须按照相应的规定进行，每个宏都有相应的宏名。

2. 汇编程序

汇编程序的功能是将用汇编语言编写的源程序翻译成机器指令程序。它一般至少需要两次扫描源程序才能完成翻译过程。第一次扫描的主要工作是定义符号的值并创建一个符号表(ST)；第二次扫描的任务是产生目标程序。除了使用前一次扫描所产生的符号表(ST)外，还要使用机器指令表(MOT2)。在第二次扫描过程中，可执行汇编语句应被翻译成对应的二进制代码机器指令。这一过程涉及两个方面的工作：一是把机器指令助记符转换成二进制机器指令操作码，这可通过查找 MOT2 来实现；二是求出操作数区各操作数的值(用二进制表示)。

2.2.1.2　编译程序的基本原理

1. 编译过程概述

编译程序的功能是把用高级语言书写的源程序翻译成与之等价的目标程序。编译过程划分成词法分析、语法分析、语义分析、中间代码生成、代码优化和目标代码生成 6 个阶段，实际的编译器可能会将其中的某些阶段结合在一起进行处理，比如说表格管理和出错处理与上述 6 个阶段都有联系。

1) 词法分析阶段

词法分析阶段的任务是对源程序从前到后(从左到右)逐个字符进行扫描，从中识别出一个个“单词”符号。“单词”符号是程序设计语言的基本语法单位，如关键词、标识符等。词法分析程序输出的“单词”常常采用二元组的方式，即单词类别和单词自身的值。

2) 语法分析阶段

语法分析的任务是在词法分析的基础上，根据语言的语法规则将单词符号序列分解成各类语法单位，如“表达式”“语句”和“程序”等。

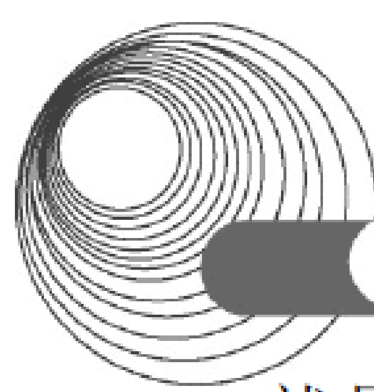
词法分析和语法分析本质上都是对源程序的结构进行分析。

3) 语义分析阶段

语义分析阶段主要是审查源程序是否存在语义错误，并收集类型信息供后面的代码生成阶段使用，只有语法和语义都正确的源程序才能翻译成正确的目标代码。语义分析的一个主要工作是进行类型分析和检查。

4) 中间代码生成阶段

中间代码是一种结构简单且含义明确的记号系统，可以有多种形式。中间代码的设计原则主要有两点：一是容易生成；二是容易被翻译成目标代码。中间代码生成阶段的工作



就是根据语义分析的输出生成中间代码。

语义分析和中间代码生成所依据的是语言的语义规则。

5) 代码优化阶段

代码优化阶段的任务是对前阶段产生的中间代码进行变换或进行改造，目的是使生成的目标代码更为高效，即省时间和省空间。优化过程可以在中间代码生成阶段进行，也可以在目标代码生成阶段进行。

6) 目标代码生成阶段

目标代码生成阶段的任务是把中间代码变换成特定机器上的绝对指令代码、可重定位的指令代码或汇编指令代码。这是编译的最后阶段，它的工作与具体的机器密切相关。

7) 符号表管理

符号表管理阶段的任务是在符号表中记录源程序中各个符号的必要信息，以辅助语义的正确性检查和代码生成。符号表的建立可以始于词法分析阶段，也可以放到语法分析阶段，但符号表的使用有时会延续到目标代码的运行阶段。

8) 出错处理

用户编写的源程序中的错误大致可分为静态错误和动态错误。动态错误也称为动态语义错误，指程序中包含的逻辑错误。静态错误是指编译阶段发现的程序错误，可分为语法错误和静态语义错误。出错处理程序的任务包括检查错误、报告出错信息、排错、恢复编译工作。

2. 文法和语言的形式描述

1) 文法的定义

描述语言语法结构的形式规则称为文法。文法 G 是一个四元组，可表示为 $G=(V_N, V_T, P, S)$，其中 V_T 是一个非空有限集，其中的每个元素称为一个终结符；V_N 是一个非空有限集，其每个元素称为非终结符。$V_N \cap V_T = \varnothing$。$P$ 是产生式的有限集合，每个产生式是形如 $\alpha \rightarrow \beta$ 的规则，其中 α 称为产生式的左部，β 称为产生式的右部。$S \in V_N$，称为开始符号，它至少要在一条产生式中作为左部出现。

2) 文法的分类

乔姆斯基把文法分成4种类型，即0型、1型、2型和3型。

0型文法也称为短语文法，其能力相当于图灵机。

(1) 1型文法也称为上下文有关文法，这种文法意味着对非终结符的替换必须考虑上下文，并且一般不允许替换成ε串，此文法对应于线性有界自动机。

(2) 2型文法是上下文无关文法，对非终结符的替换无须考虑上下文，它对应于下推自动机。

(3) 3型文法等价于正规式，因此也称为正规文法或线性文法，它对应于有限状态自动机。

3) 句子和语言

设有文法 $G=(V_N, V_T, P, S)$

(1) 推导和直接推导。从文法的开始符号 S 出发，反复使用产生式，将产生式左部的非终结符替换为右部的文法符号序列，直至产生一个终结符的序列时为止。若有产生式 $\alpha \rightarrow \beta \in P$，$\gamma, \delta \in V^*$，则 $\gamma\alpha\delta \Rightarrow \gamma\beta\delta$ 称为文法 G 中的一个直接推导。

(2)　直接归约和归约。若文法 G 中有一个直接推导 $\alpha \Rightarrow \beta$，则称 α 是 β 的一个直接归约；若文法 G 中有一个推导 $\gamma \overset{*}{\underset{G}{\Rightarrow}} \delta$，则称 γ 是 δ 的一个归约。

(3)　句型和句子。若文法 G 的开始符号为 S，那么，从开始符号 S 能推导出的符号串称为文法的一个句型，即 α 是文法 G 的一个句型，当且仅当有以下推导 $S \overset{*}{\underset{G}{\Rightarrow}} \alpha$，$\alpha \in V^*$，若 X 是文法 G 的一个句型，且 $X \in V_T^*$，则称 X 是文法 G 的一个句子。

(4)　语言。从文法 G 的开始符号出发，所能推导出的句子的全体称为文法 G 产生的语言，记为 $L(G)$。

(5)　文法的等价。若文法 G_1 与文法 G_2 产生的语言相同，即 $L(G_1)=L(G_2)$，则称这两个文法是等价的。

3.　词法分析

1)　正规表达式和正规集

对于字母表 Σ，其上的正规表达式(也称正则表达式，简称正规式)及其表示的正规集可以递归定义如下。

(1)　ε 是一个正规式，它表示集合 $L(\varepsilon)=\{\varepsilon\}$。

(2)　若 a 是 Σ 上的字符，则 a 是一个正规式，它所表示的正规集为 $\{a\}$。

(3)　若正规式 r 和 s 分别表示正规集 $L(r)$ 和 $L(s)$，则

①　$r|s$ 是正规式，表示集合 $L(r) \cup L(s)$。

②　$r \cdot s$ 是正规式，表示集合 $L(r)L(s)$。

③　r^* 是正规式，表示集合 $(L(r))^*$。

④　(r) 是正规式，表示集合 $L(r)$。

仅由有限次地使用上述 3 个步骤定义的表达式才是 Σ 上的正规式，其中运算符“|”“·”“*”分别称为“或”“连接”和“闭包”。若两个正规式表示的正规集相同，则认为两者等价。

2)　有限自动机

有限自动机是一种识别装置的抽象概念，它能够正确地识别正规集。

(1)　确定的有限自动机。

一个确定的有限自动机(DFA)是个五元组：(S, Σ, f, s_0, Z)，其中：

①　S 是一个有限集，其每个元素称为一个状态。

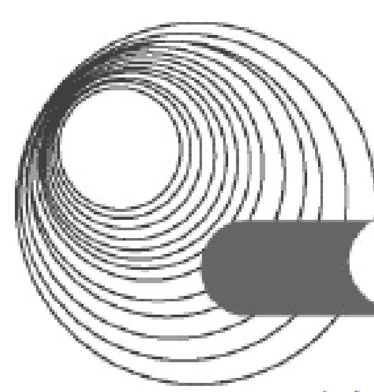
②　Σ 是一个有限字母表，其每个元素称为一个输入字符。

③　f 是从 $S \times \Sigma \rightarrow S$ 上的单值部分映像。

④　$s_0 \in S$ 是唯一的一个开始状态。

⑤　Z 是非空的终止状态集合。

一个 DFA 可以用两种直观的方式表示，即状态转换图和状态转换矩阵。状态转换图简称为转换图，它是一个有向图。DFA 中的每个状态对应转换图中的一个节点，DFA 中的每个转换函数对应图中的一条有向弧，若转换函数为 $f(A, a)=Q$，则该有向弧从节点 A 出发，进入节点 Q，字符 a 是弧上的标记。状态转换矩阵可以用一个二维数组 M 表示，矩阵元素的行下标表示状态，列下标表示输入字符，$M[A,a]$ 的值是当前状态为 A、输入为 a 时应转换



到的下一状态。在转换矩阵中，一般以第一行的行下标所对应的状态作为初态，而终态则需要特别指出。

(2) 不确定的有限自动机。

一个不确定的有限自动机(NFA)也是一个五元组，它与确定的有限自动机的区别如下。

① f 是从 $S\times\Sigma\rightarrow 2^S$ 上的映像。对于 S 中的一个给定状态及输入符号，返回一个状态的集合。

② 有向弧上的标记可以是ε。

显然，DFA 是 NFA 的特例。

实际上，对于每个 NFA M，都存在一个 DFA N，且 $L(M)=L(N)$。

对于任何两个有限自动机 M_1 和 M_2，如果 $L(M_1)=L(M_2)$，则称 M_1 和 M_2 是等价的。

3) NFA 到 DFA 的转换

设 NFA $N=(S, \Sigma, f, s_0, Z)$，与之等价的 DFA $M=(S', \Sigma, f', q_0, Z')$，用子集法将非确定的有限自动机确定化的算法步骤如下。

(1) 求出 DFA M 的初态 q_0，此时 S'仅含初态 q_0，并且没有标记。

(2) 对于 S'中尚未标记的状态 $q_i=\{s_{i1},s_{i2},\cdots,s_{im}\}$和 $s_{ij}\in S(j=1,2,\cdots,m)$进行下述处理。

① 标记 q_i。

② 对于每个 $a\in\Sigma$，令 $T=f(s_{i1},s_{i2},\cdots,s_{im},a)$，$q_j=\varepsilon_\text{CLOSURE}(T)$。

③ 若 q_i 尚不在 S'中，则将 q_j 作为一个未加标记的新状态添加到 S'，并把状态转换函数 $f'(q_i,a)=q_j$ 添加到 DFA M。

(3) 重复步骤(2)，直到 S'中不再有未标记的状态时为止。

(4) 令 $Z'=\{q|q\in S'且 q\cap Z\neq\varnothing\}$。

注：若 I 是 NFA N 的状态集合的一个子集，其中 $\varepsilon_\text{CLOSURE}(I)$ 的定义如下。

① 状态集 I 的 $\varepsilon_\text{CLOSURE}(I)$ 是一个状态集。

② 状态集 I 的所有状态属于 $\varepsilon_\text{CLOSURE}(I)$。

③ 若 s 在 I 中，那么从 s 出发经过任意条 ε 弧到达的状态 s'都属于 $\varepsilon_\text{CLOSURE}(I)$。

从 NFA 转换得到的 DFA 不一定是最简化的，可以通过等价变换将 DFA 进行最小化处理。

4) 正规式与有限自动机之间的转换

(1) 对于Σ上的 NFA M，可以构造一个Σ上的正规式 R，使得 $L(R)=L(M)$。

构造过程分以下两步进行。

① 在 M 的状态转换图中加两个节点 x 和 y。

② 按图 2-1 所示的方法逐步消去 M 中的除 x 和 y 的所有节点。

(2) 对于 Σ 上的每一个正规式 R，可以构造一个 Σ 上的 NFA M，使得 $L(M)=L(R)$。

(3) 构造过程分两步进行。

① 对于正规式 R，可用如图 2-2 所示的拓广状态图表示。

② 通过对正规式 R 进行分裂并加入新的节点，逐步把图转变成每条弧上的标记是 Σ 上的一个字符或ε，转换规则如图 2-3 所示。

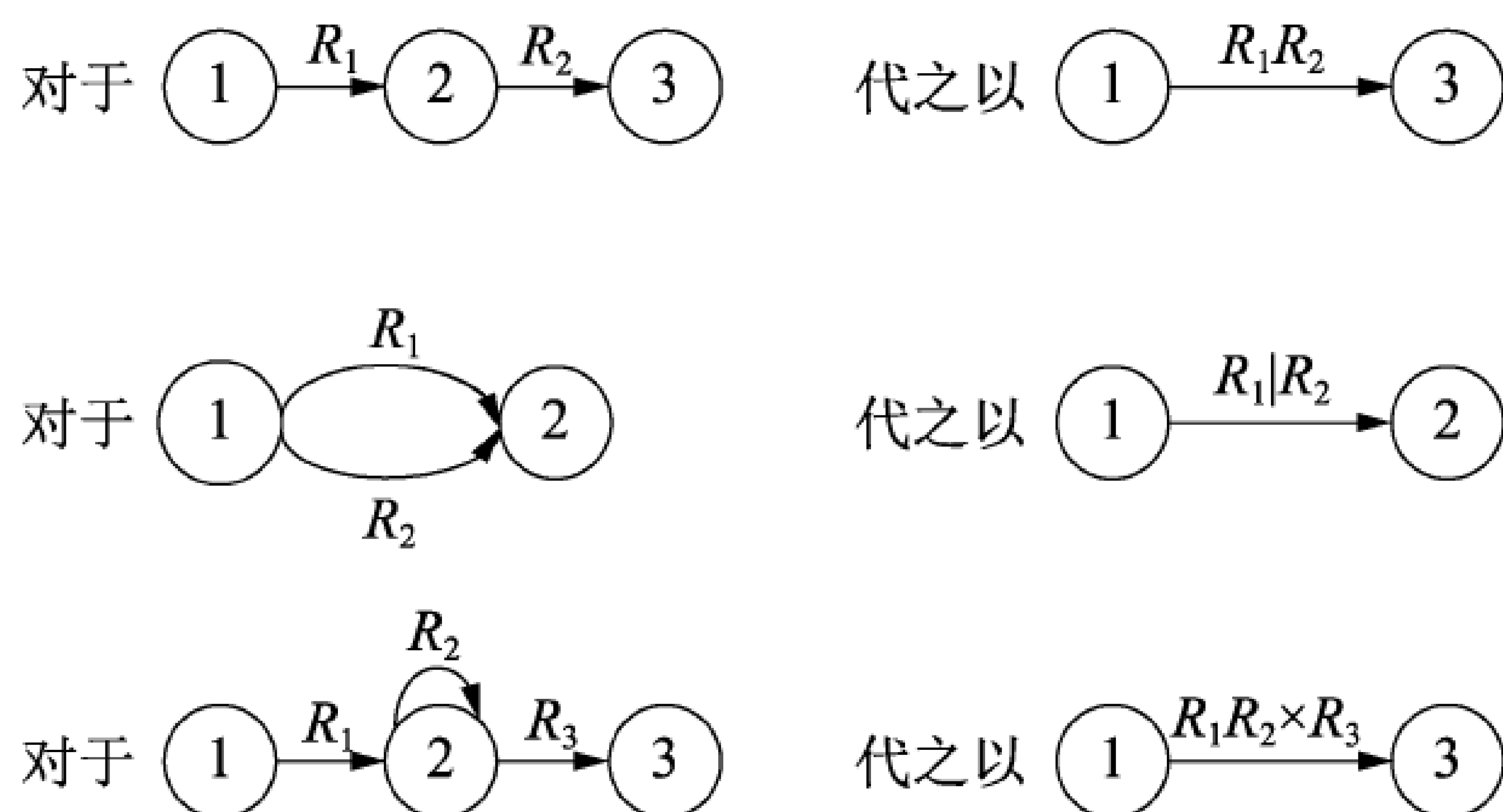


图 2-1　状态转换图(消去中间节点)

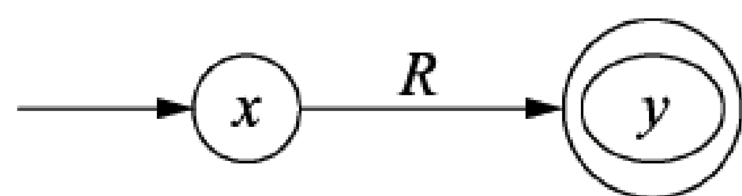


图 2-2　拓广状态图

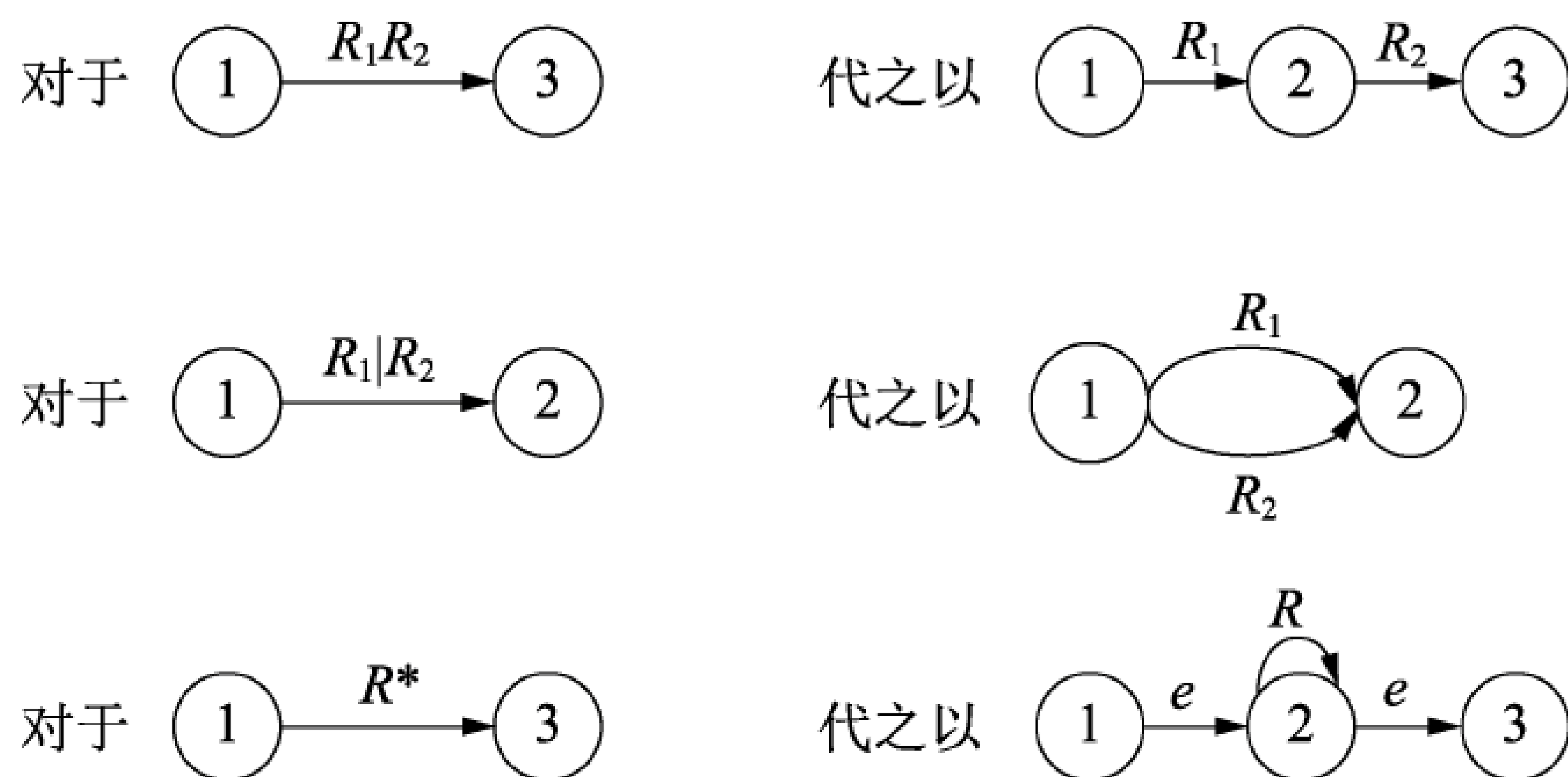


图 2-3　状态转换图(加入新节点)

5)　词法分析器的构造

词法分析器的构造过程如下。

(1)　用正规式描述语言中的单词构成规则。

(2)　为每个正规式构造一个 NFA，用于识别正规式所表示的正规集。

(3)　将构造出的 NFA 转换成等价的 DFA。

(4)　对 DFA 进行最小化处理，使其最简。

(5)　根据 DFA 构造词法分析器。

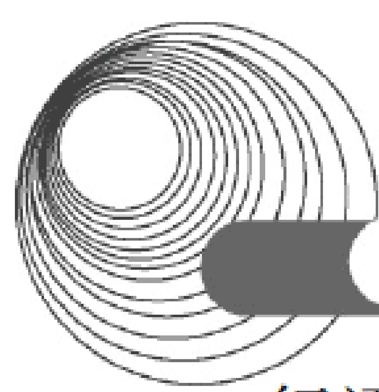
4.　语法分析

语法分析的任务是根据语言的语法规则，分析单词串是否构成短语和句子，同时检查和处理程序中的语法错误。根据产生语法树的方向，语法分析可分为自底向上和自顶向下两类。

自顶向下的分析是对给定的符号串，试图自顶向下地为其构造一棵语法树，或者说从文法的开始符号出发，为其构造一个最佳推导。

自底向上的分析是对给定的符号串，试图自底向上地为其构造一棵语法树，或者说从给定的符号串本身出发，试图将其归约为文法的开始符号。

算符优先文法属于自底向上的分析法，它利用各个算符间的优先关系和结合规则来进



行语法分析，特别是用于分析各种表达式。算符优先文法的任何产生式的右部都会出现两个非终结符相邻的情况，且任何一对终结符之间至多只有 3 种算符关系，即“>”“<”和“=”之一成立。

5. 中间代码优化

1) 局部优化

局部优化是在基本块上的优化。基本块是指程序中一个顺序执行的语句序列，其中只有一个入口和一个出口。划分基本块的方法如下。

(1) 求出四元式程序中各个基本块的入口语句。

(2) 对每一入口语句，构造其所属的基本块。它是由该语句到下一入口语句(不包括下一入口语句)，或到一条转移语句(包括该转移语句)，或到一条停语句(包括该停语句)之间的语句序列组成的。

(3) 凡未被纳入某一基本块的语句，都是程序中控制流程无法到达的语句，因而也是不会被执行到的语句，因此可以把它们删除。

一个基本块可以用一个 DAG(有向无环)图表示。在一个基本块内，通常可进行以下 3 种优化，即合并已知量、删除无用赋值和删除多余运算。

2) 控制流图和循环优化

(1) 控制流图。

一个程序的控制流图是一个有向图，其节点是程序中的基本块，它有唯一的首节点，即包含程序第一条语句的基本块，从首节点出发，到控制流图中的每个节点都存在路径。

由程序的各基本块构造相应控制流图的方法是：对于程序中的两个基本块 B_i 和 B_j，若 B_j 紧接着 B_i 被执行，则从 B_i 引一条有向边到 B_j，称 B_i 是 B_j 的直接前驱，B_j 是 B_i 的直接后继。

(2) 循环优化。

循环就是控制流图中具有唯一入口节点的强连通子图，从循环外进入循环时，必须首先经过循环的入口节点。

基于循环的优化处理有代码外提、强度削弱、删除归纳变量等。

6. 目标代码生成

代码生成器以经过语义分析或优化后的中间代码为输入，以特定的机器语言或汇编代码为输出。代码生成主要考虑以下问题，即中间代码形式、目标代码形式、寄存器的分配、计算次序的选择。

2.2.1.3 解释程序的基本原理

解释程序是一种语言处理程序，在词法、语法和语义分析方面与编译程序的工作原理基本相同，但在运行用户程序时，它直接执行源程序或源程序的内部形式(中间代码)。因此，解释程序并不产生目标程序，这是它和编译程序的主要区别。

解释程序的结构通常可以分成两部分：第一部分是分析部分，包括通常的词法分析、语法分析和语义分析程序，经语义分析后把源程序翻译成中间代码，中间代码常采用逆波兰表示形式；第二部分是解释部分，用来对第一部分产生的中间代码进行解释执行。

2.2.2　典型例题分析

例 1　C 程序中全局变量的存储空间在__(22)__分配。(2015 年上半年试题 22)

(22) A. 代码区　　B. 静态数据区　　C. 栈区　　D. 堆区

解析：代码区：存放函数体的二进制代码。

栈区：由编译器自动分配释放，存放函数的参数值、局部变量值等。

堆区：一般由程序员分配释放， 若程序员不释放，程序结束时可能由操作系统回收。

静态数据区：内存在程序启动的时候才被分配，而且可能直到程序开始执行的时候才被初始化，所分配的内存在程序的整个运行期间都存在，如全局变量、static 变量等。

答案：B

例 2　以下关于语言 $L=\{a^nb^n|n>1\}$的叙述中，正确的是__(48)__。(2013 年上半年试题 48)

(48) A. 可用正规式“$aa*bb*$”描述，但不能通过有限自动机识别

B. 可用正规式“a^mb^m”表示，也可用有限自动机识别

C. 不能用正规式表示，但可以用有限自动机识别

D. 不能用正规式表示，也不能通过有限自动机识别

解析：根据正规式和有限自动机的规范，结合题中给出的语言 L，很明显都无法将语言 L 表示和识别出来。

答案：D

例 3　编译过程中，对高级语言程序语句的翻译主要考虑声明语句和可执行语句。对声明语句，主要是将需要的信息正确地填入合理组织的__(49)__中；对可执行语句，则是__(50)__。(2013 年上半年试题 49、50)

(49) A. 符号表　　B. 栈　　C. 队列　　D. 树

(50) A. 翻译成机器代码并加以执行　　B. 转换成语法树

C. 翻译成中间代码或目标代码　　D. 转换成有限自动机

解析：编译程序的功能是把用高级语言书写的源程序翻译成与之等价的目标程序(汇编语言或机器语言)。编译程序的工作过程可以分为 6 个阶段，即词法分析、语法分析、语义分析、中间代码生成、代码优化、目标代码生成，实际的编译器中可能会将其中的某些阶段结合在一起进行处理。各个阶段逻辑上可以划分为前端和后端两部分。前端包括从词法分析到中间代码生成各个阶段的工作，后端包括中间代码优化、目标代码生成与优化等阶段。这样，以中间代码为分水岭，把编译器分成了与机器有关的部分和与机器无关的部分。符号表的作用是记录源程序中各个符号的必要信息，以辅助语义的正确性检查和代码生成，在编译过程中需要对符号表进行快速有效地查找、插入、修改和删除等操作。

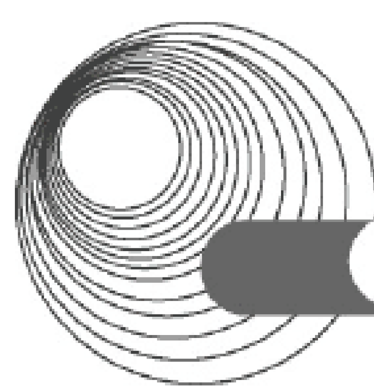
答案：(49) A　　(50) C

例 4　语言 $L=\{a^mb^n|m\geq 0, n\geq 1\}$的正规表达式是__(21)__。(2012 年下半年试题 21)

(21) A. aa^*bb^*　　B. a^*bb^*　　C. aa^*b^*　　D. a^*b^*

解析：$m\geq 0$，a^m说明可以有 0 个 a 或者多个 a；$n\geq 1$，b^n说明至少有一个 b 或者多个 b。a^*表示由 0 个或者多个 a 构成的集合，可以表示$\{a^m| m\geq 0\}$；b^*表示由 0 个或者多个 b 构成的集合，bb^*才可以表示$\{b^n| n\geq 1\}$。

答案：B



例 5 算术表达式“(a-b)*(c+d)”的后缀式是__(21)__。(2014 年下半年试题 21)

(21) A．ab-cd+* B．abcd-*+ C．ab-*cd+ D．ab-c+d*

解析：后缀式是波兰逻辑学家卢卡西维奇发明的一种表达方式，把运算符号写在运算对象的后面，如把 a+b 写成 ab+，这种表示法的优点是根据运算对象和算符的出现次序进行计算，不需要使用括号。

答案：A

例 6 某程序运行时陷入死循环，则可能的原因是程序中存在__(48)__。(2015 年下半年试题 48)

(48) A．词法错误 B．语法错误
C．动态的语义错误 D．静态的语义错误

解析：死循环错误属于典型的语义错误，但静态的语义错误可被编译器发现，到程序真正陷入死循环说明编译器并未发现，所以属于动态语义错误。

答案：C

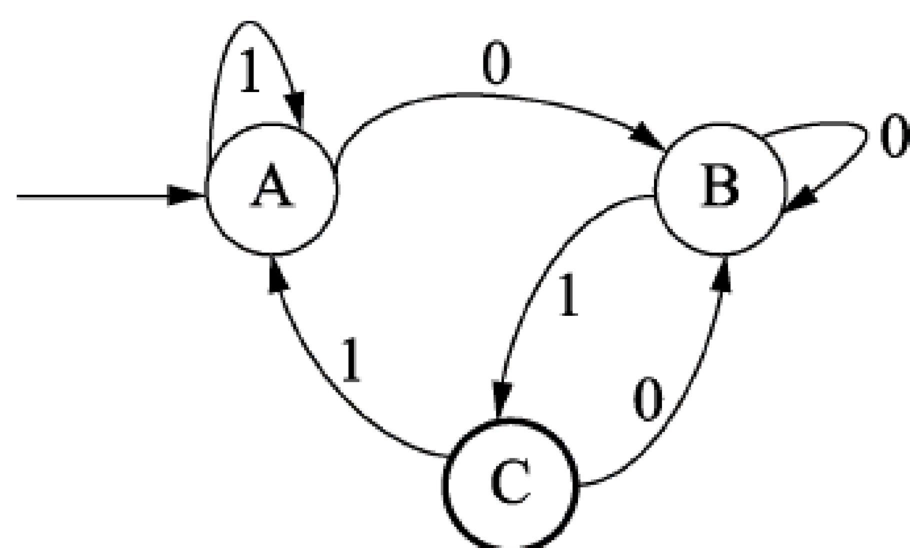
例7 编译程序对高级语言源程序进行编译的过程中，要不断收集、记录和使用源程序中一些相关符号的类型和特征等信息，并将其存入__(22)__中。(2014年上半年试题22)

(22) A．符号表 B．哈希表 C．动态查找表 D．栈和队列

解析：编译过程中编译程序不断汇集和反复查证出现在源程序中各种名字的属性和特征信息等有关信息。这些信息通常记录在一张或几张符号表中。符号表的每一项有两部分：一部分是名字(标识符)；一部分是名字属性(标识符的有关信息)。编译过程中，每当扫描器(词法分析器)识别出一个名字后，编译程序就查阅符号表，看其是否在符号表中。符号表在编译全过程的地位和作用非常重要，是进行上下文合法性检查和语义处理及代码生成的依据。符号表总体结构的设计和实现是与源语言的复杂性(包括词法结构、语法结构的复杂性)有关，还与对于编译系统在时间效率和空间效率方面的要求有关。

答案：A

例 8 下图所示为一个有限自动机(其中，A 是初态、C 是终态)，该自动机所识别的字符串的特点是__(48)__。(2012 年上半年试题 48)

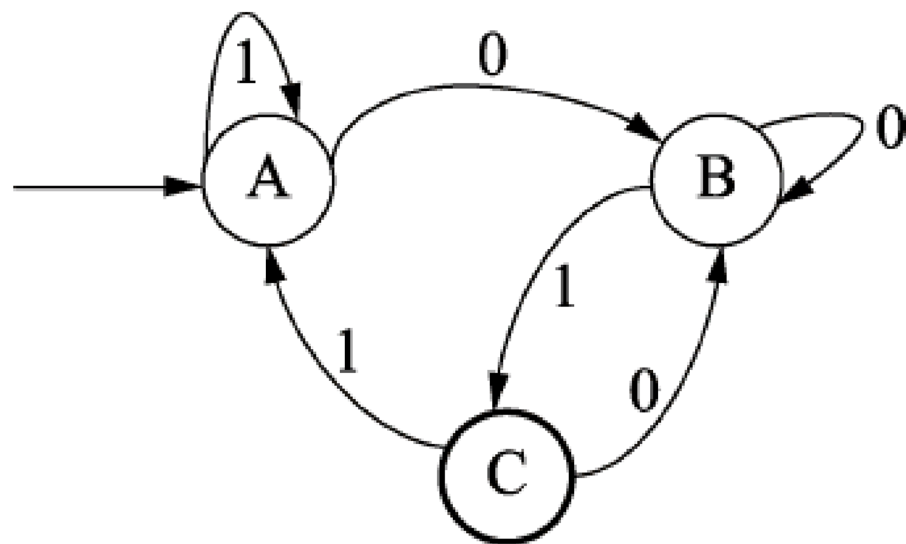


(48) A. 必须以 11 结尾的 0、1 串 B. 必须以 00 结尾的 0、1 串
C. 必须以 01 结尾的 0、1 串 D. 必须以 10 结尾的 0、1 串

解析：有限自动机可识别的字符串，是指从有限自动机的初态出发，存在一条到达终态的路径，其上的标记所构成的字符串。本题 C 是终态，C 的前一状态只能是 B，由 B 到 C 输入的是 1；B 的前一状态可以是 A，也可以是 B，还可以是 C，但输入的都是 0。可见，该有限自动机识别的串必须以 01 结尾。

答案：C

例 9　下图所示为一个有限自动机(其中，A 是初态、C 是终态)，该自动机识别的语言可用正规式 (48) 表示。(2011 年下半年试题 48)



(48) A. (0|1)*01　　B. 1*0*10*1　　C. 1*(0)*01　　D. 1*(0|10)*1*

解析： 在正规式中，符号*表示重复若干次(包括 0 次)，符号|表示“或”。在状态 A，可以输入 1 或 0，如果输入 1 还可以回到状态 A，如果输入 0 则直接到达状态 B；在状态 B，可以输入 0 或 1，如果输入 0 则还回到状态 B，而输入 1，则进入状态 C；在状态 C 可以输入 0 或 1，输入 0 到达状态 B，输入 1 到达状态 A，但由于 C 是终态，自动机可识别的语言是由 0、1 构成的字符串的集合，但该集合必须以 01 结束，因此选项 A 正确。

答案： A

例 10　表达式采用逆波兰式表示时，利用 (22) 进行求值。(2015 年下半年试题 22)

(22) A. 栈　　B. 队列　　C. 符号表　　D. 散列表

解析： 逆波兰将运算符写在操作数之后，因此也称为后缀表达式。将一个普通的中序表达式转换为逆波兰表达式的一般过程是：从左至右扫描表达式，如果当前字符为变量或者为数字，则入栈；如果是运算符，则将栈顶两个元素弹出做相应运算，结果再入栈；以此类推，最后当表达式扫描完，栈里的就是结果。

答案： A

例 11　若一种程序设计语言规定其程序中的数据必须具有类型，则有利于 (22) 。(2011 年上半年试题 22)

① 在翻译程序的过程中为数据合理分配存储单元
② 对参与表达式计算的数据对象进行检查
③ 定义和应用动态数据结构
④ 规定数据对象的取值范围及能够进行的运算
⑤ 对数据进行强制类型转换

(22) A. ①②③　　B. ①②④　　C. ②④⑤　　D. ③④⑤

解析： 数据是程序操作的对象，类型说明数据占用的内存和存放形式。数据类型不仅可用于在基础机器中完成对值的布局，还可以用于检查表达式中对运算的应用是否正确。

答案： B

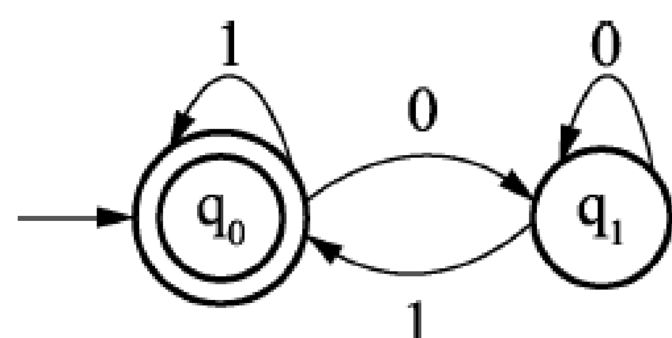
例 12　对高级语言源程序进行编译或解释的过程可以分为多个阶段，解释方式不包含 (48) 阶段。(2015 年上半年试题 48)

(48) A. 词法分析　　B. 语法分析　　C. 语义分析　　D. 目标代码生成

解析： 解释方式运行用户程序时，直接执行源程序或者源程序的中间表示形式，不产生源程序的目标程序。

答案： D

例 13 某非确定的有限自动机(NFA)的状态转换图如下图所示(q_0既是初态也是终态)。以下关于该 NFA 的叙述中，正确的是__(49)__。(2015 年下半年试题 49)



(49) A．其可识别的 0、1 序列的长度为偶数
B．其可识别的 0、1 序列中 0 与 1 的个数相同
C．其可识别的非空 0、1 序列中开头和结尾字符都是 0
D．其可识别的非空 0、1 序列中结尾字符是 1

解析：在初态 q_0 输入 1 又回到 q_0，输入 0 则迁移到状态 q_1，可见在 0 之前可以有任意多个 1(1*)；在 q_1 状态输入 0，则回到 q_1，输入 1 则迁移到终态 q_0，因此在 1 之前可以有任意多个 0(0*)，因此可识别的字符串为 1*00*1。

答案：D

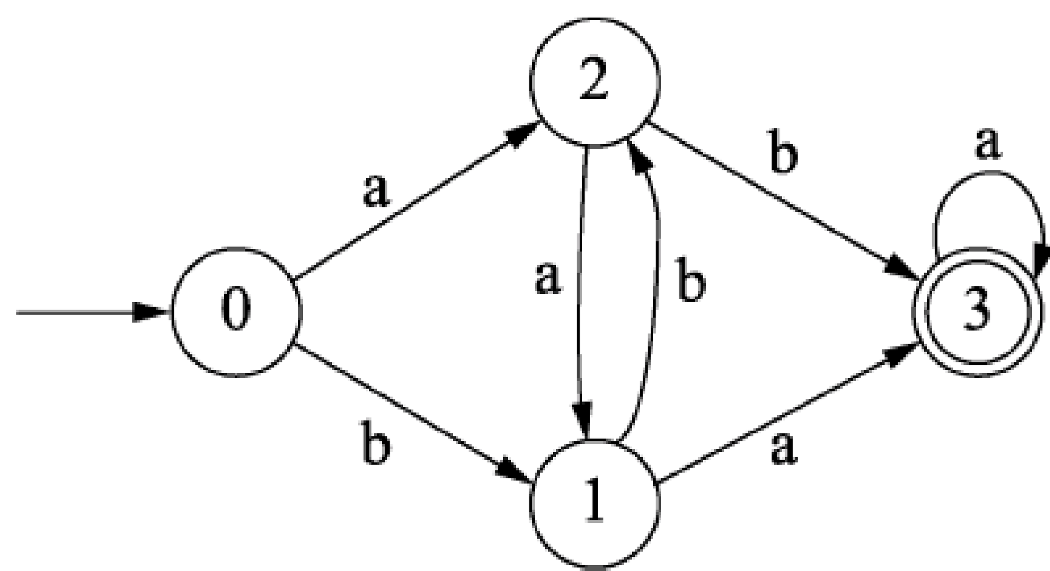
例 14 对高级语言源程序进行编译的过程可以分为多个阶段，分配寄存器的工作在__(48)__阶段进行。(2014 年下半年试题 48)

(48) A．词法分析　　B．语法分析　　C．语义分析　　D．目标代码生成

解析：目标代码生成阶段应考虑直接影响到目标代码速度的 3 个问题：一是如何生成较短的目标代码；二是如何充分利用计算机中的寄存器，减少目标代码访问存储单元的次数；三是如何充分利用计算机指令系统的特点，以提高目标代码的质量。

答案：D

例 15 下图所示的有限自动机中，0 是初始状态，3 是终止状态，该自动机可以识别__(22)__。(2010 年下半年试题 22)

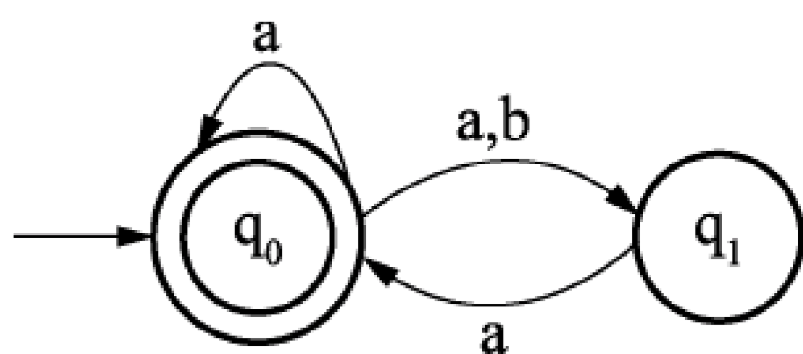


(22) A．abab　　B．aaaa　　C．bbbb　　D．abba

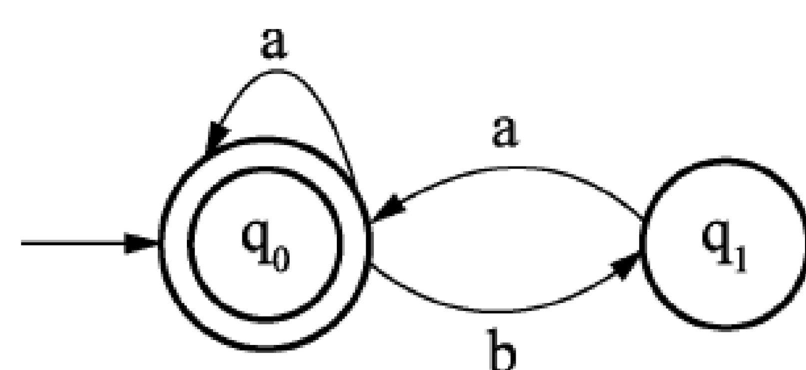
解析：从初始状态到终止状态有多条路径。在状态 0 输入 a 到达状态 2，在状态 2 可输入 a 或 b，输入 a 到达状态 1，输入 b 到达状态 3，状态 3 下输入 a 还回到状态 3；在状态 1 可输入 a 或 b，输入 a 到达状态 3，输入 b 到达状态 2。

答案：B

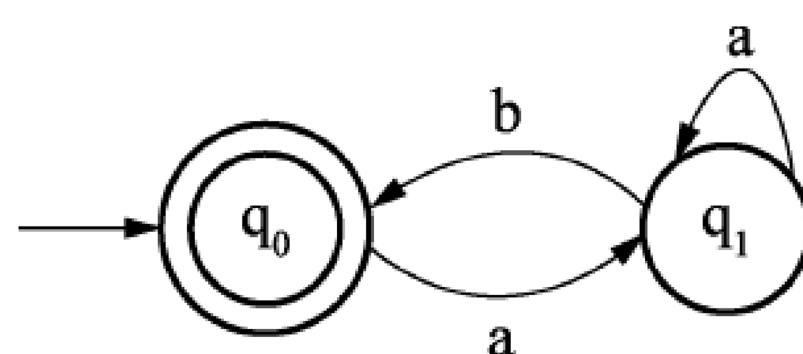
例 16 某非确定的有限自动机(NFA)的状态转换图如下图所示(q_0既是初态也是终态)，与该 NFA 等价的确定的有限自动机(DFA)是__(49)__。(2015 年上半年试题 49)



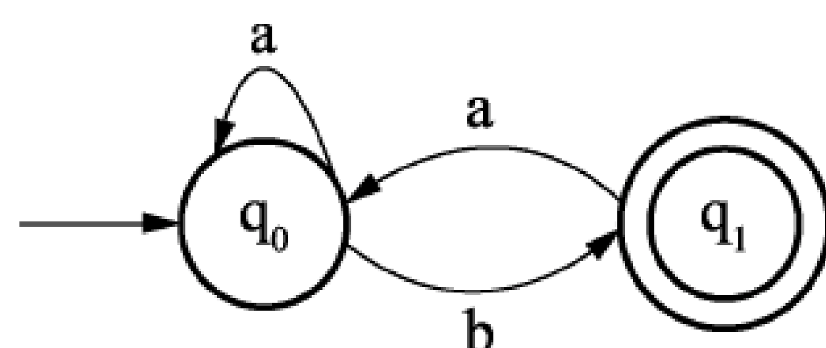
(49) A.



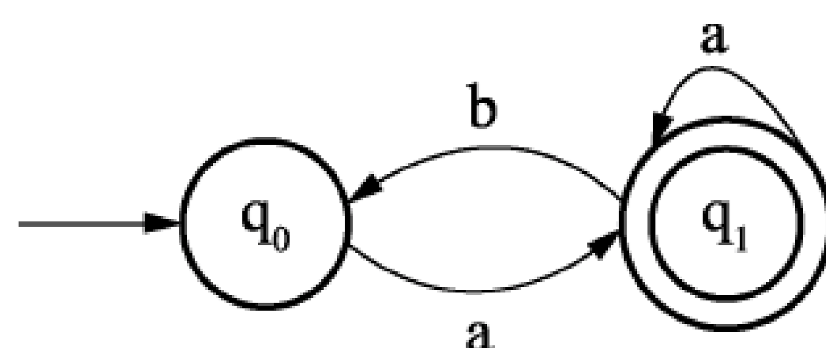
B.



C.



D.



解析：一个非确定自动机(NFA)在读入符号串之后，并不确切地知道自动机处于哪个状态。但可以肯定一定处于状态集中的某一状态。该状态集记作{$q_1,q_2,\cdots,q_k$}。而一个等价的确定自动机(DFA)读入同样的 w 一定处于某个确定的状态上。这样，都是读入同样的 w，DFA 到达某一个状态，而 NFA 到达某一个状态集。由 w 的任意性，可将 NFA 的所有的状态集和 DFA 的状态一一对应起来。这种对应的前提就是能识别同样的输入串。

答案：A

例 17　以下关于汇编语言的叙述中，错误的是 (50) 。(2010 年下半年试题 50)

(50) A．汇编语言源程序中的指令语句将被翻译成机器代码

B．汇编程序先将源程序中的伪指令翻译成机器代码，然后再翻译指令语句

C．汇编程序以汇编语言源程序为输入，以机器语言表示的目标程序为输出

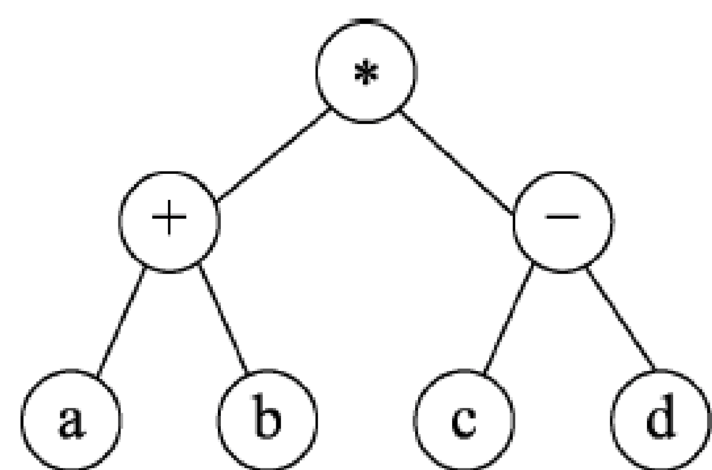
D．汇编语言的指令语句必须具有操作码字段，可以没有操作数字段

解析：汇编程序的功能是将汇编语言所编写的源程序翻译成机器指令程序。汇编语言源程序语句可分为指令语句、伪指令语句和宏指令语句。指令语句汇编后产生相应的机器代码；伪指令语句指示汇编程序在汇编源程序时完成某些操作，汇编后不产生机器代码。

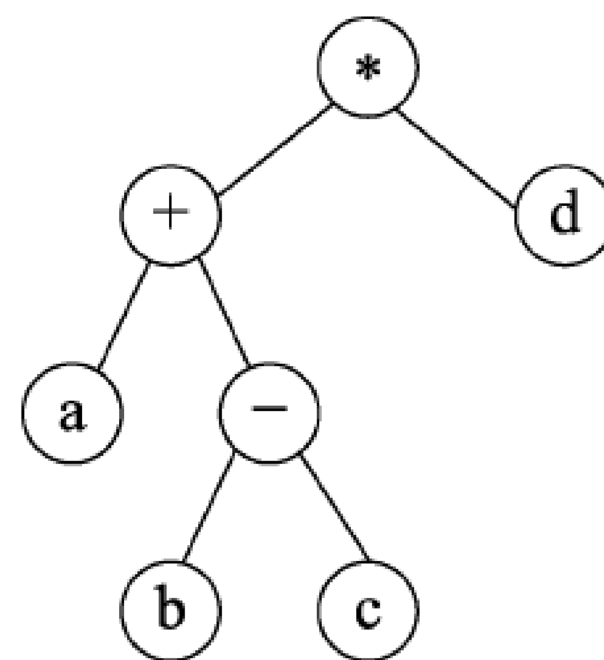
答案：B

例 18　与算术表达式“(a+(b-c))*d”对应的树是(21)(2015 年上半年试题 21)

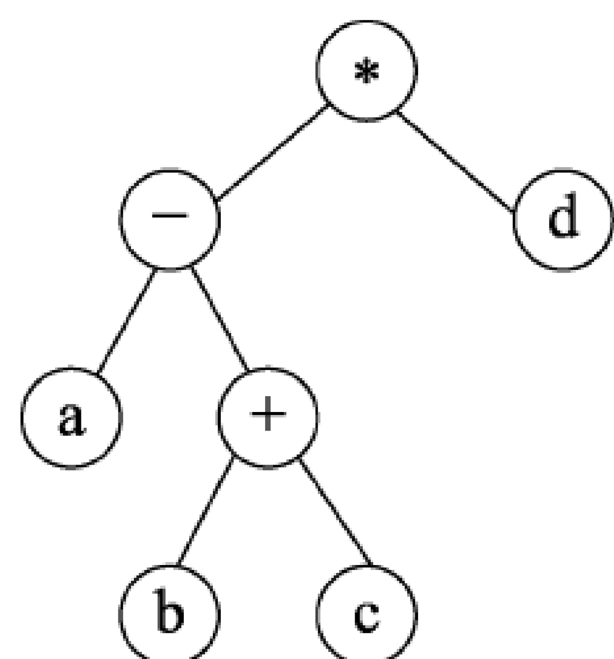
(21) A.



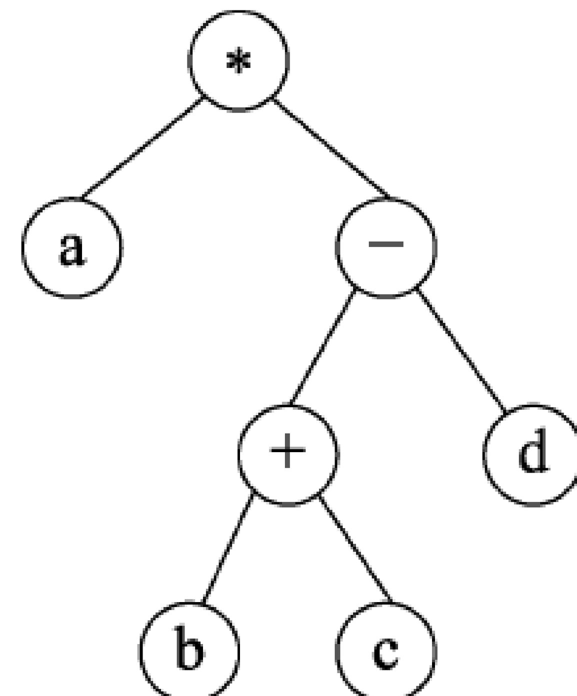
B.

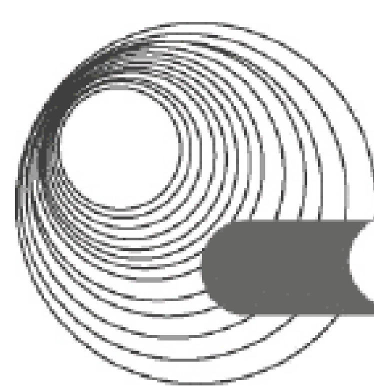


C.



D.





解析：表达式用树形来表示时通常采用中缀形式，运算符在树中放在非终端节点的位置上，操作数放在叶子节点处。处理时，首先找到运算级别最低的运算符“*”作为根节点，继而确定该根节点的左、右子树节点在表达式串中的范围为 a+(b-c)和 d；再在对应的范围内寻找运算级别最低的运算符作为子树的根节点，直到范围内无运算符，则剩余的变量或数为表达式树的叶子。

答案：B

例 19　递归下降分析方法是一种(50)方法。(2015 年上半年试题 50)

(50) A．自底向上的语法分析　　B．自上而下的语法分析
C．自底向上的词法分析　　D．自上而下的词法分析

解析：在编译方式下，机器上运行的是与源程序等价的目标程序，源程序和编译程序都不再参与目标程序的执行过程；而在解释方式下，解释程序和源程序要参与到程序的运行过程中，运行程序的控制权在解释程序。解释器翻译源程序时不产生独立的目标程序，而编译器则需要将源程序翻译成独立的目标程序。

答案：A

例20　以下关于实现高级程序设计语言的编译和解释方式的叙述中，正确的是 (48) 。(2014年上半年试题48)

(48) A．在编译方式下产生源程序的目标程序，在解释方式下不产生
B．在解释方式下产生源程序的目标程序，在编译方式下不产生
C．编译和解释方式都产生源程序的目标程序，差别是优化效率不同
D．编译和解释方式都不产生源程序的目标程序，差别在是否优化

解析：在编译方式下，机器上运行的是与源程序等价的目标程序，源程序和编译程序都不再参与目标程序的执行过程；而在解释方式下，解释程序和源程序要参与到程序的运行过程中，运行程序的控制权在解释程序。解释器翻译源程序时不产生独立的目标程序，而编译器则需要将源程序翻译成独立的目标程序。

答案：A

例 21　对于正规式 0*(10*1)*0*，其正规集中字符串的特点是 (50) 。(2010 年上半年试题 50)

(50) A．开头和结尾必须是 0　　B．1 必须出现偶数次
C．0 不能连续出现　　D．1 不能连续出现

解析：闭包运算符“*”将其运算对象进行若干次连接，因此 0*表示若干个 0 构成的串，而(10*1)*则表示偶数个 1 构成的串。

答案：B

例 22　将高级语言源程序翻译成机器语言程序的过程，常引入中间代码。以下关于中间代码的叙述中，不正确的是 (22) 。(2014 年下半年试题 22)

(22) A．中间代码不依赖于具体的机器。
B．使用中间代码可提高编译程序的可移植性
C．中间代码可以用树或图表示
D．中间代码可以用栈和队列表示

解析：中间代码是源程序的一种内部表示，或称中间语言。中间代码的作用是可使编

译程序的结构在逻辑上更为简单明确，使用中间代码可提高编译程序的可移植性，常见的有逆波兰记号、四元式、三元式和树。

答案：D

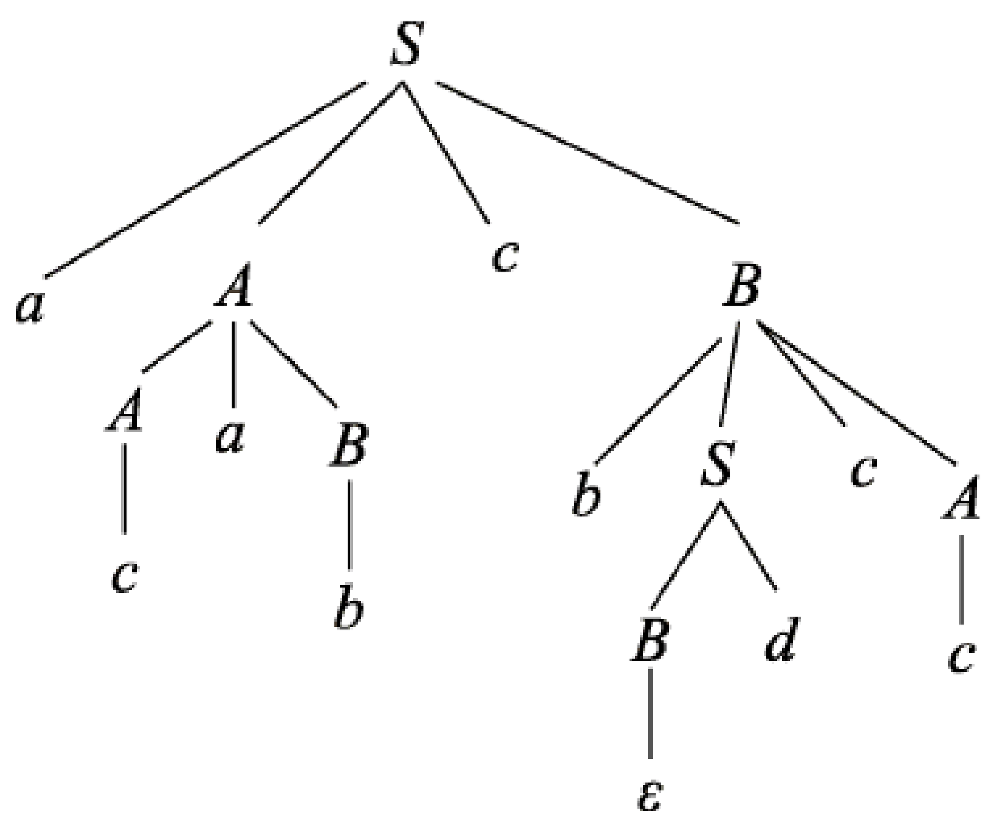
例 23　以下关于编译系统对某高级语言进行翻译的叙述中，错误的是(48)。(2009 年下半年试题 48)

(48) A．词法分析将把源程序看做一个线性字符序列进行分析
　　B．语法分析阶段可以发现程序中所有的语法错误
　　C．语义分析阶段可以发现程序中所有的语义错误
　　D．目标代码生成阶段的工作与目标机器的体系结构相关

解析：在词法分析阶段，源程序可以简单地被看成一个多行的字符串。这一阶段的任务是对源程序从前到后(从左到右)逐个字符进行扫描，从中识别出一个个“单词”符号，语法分析的任务是在词法分析的基础上，根据语言的语法规则将单词符号序列分解为各类语法单位，检查其中的语法错误；语义分析阶段的主要任务是检查源程序是否包含静态语义错误，并收集类型信息供后面的代码生成阶段使用；目标代码生成是编译器工作的最后一个阶段，这一阶段的任务是把中间代码变化为特定机器上的绝对指令代码、可重定位的指令代码或汇编指令代码，这个阶段的工作与具体的机器密切相关。因此说法 C 中发现所有语义错误是不对的。

答案：C

例 24　由某上下文无关文法 $M[S]$推导出某句子的分析树如下图所示，则错误的叙述是(50)。(2009 年下半年试题 50)

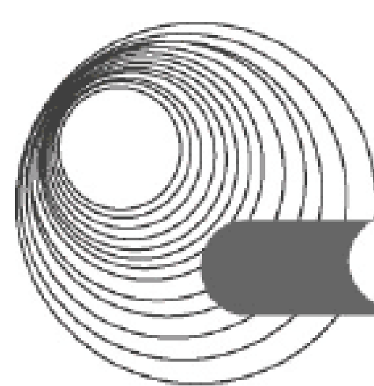


(50) A．该文法推导出的句子必须以 *a* 开头
　　B．*acabcbdcc* 是该文法推导出的一个句子
　　C．$S\rightarrow aAcB$ 是该文法的一个产生式
　　D．*a*、*b*、*c*、*d* 属于该文法的终结符号集

解析：上图所示为某上下文无关文法 $M[S]$推导出某句子的分析树，通过观察上图，只要稍作推导就可推出 *acabcbdcc* 是该文法推导出的一个句子；看该分析树的第一层分支即可知 $S\rightarrow aAcB$ 是该文法的一个产生式；而 *a*、*b*、*c*、*d* 因为在图中是分析树的叶子，所以都是该文法的终结符号；右边的 *B* 分支下有 $S\rightarrow Bd$、$B\rightarrow \varepsilon$，所以该文法推导出的句子不一定是 *a* 开头的。

答案：A

例 25　以下程序设计语言中，(20)更适合用来进行动态网页处理。(2014 年上半年试题 20)



(20) A .HTML　　B. LISP　　C. PHP　　D. JAVA/C++

解析：HTML用于处理静态网页；LISP是一种基于λ演算的函数式编程语言。

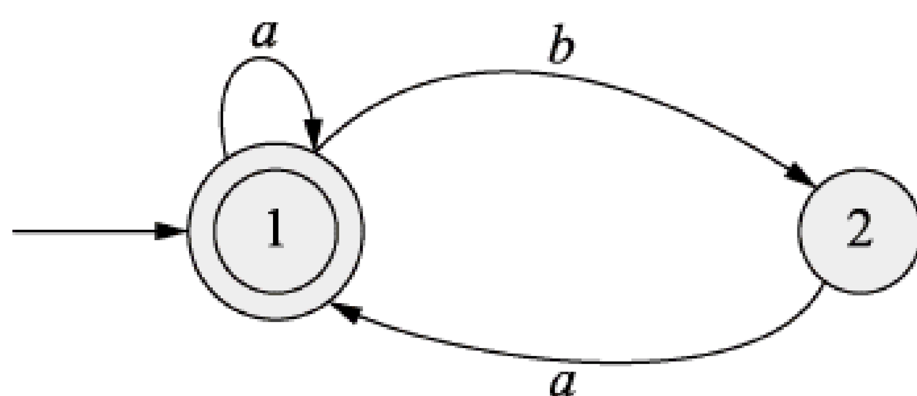
PHP 是一种通用开源脚本语言。语法吸收了 C 语言、Java 和 Perl 的特点，利于学习，使用广泛，主要适用于 Web 开发领域。它可以比 CGI 或者 Perl 更快速地执行动态网页。用 PHP 做出的动态页面与其他的编程语言相比，PHP 是将程序嵌入到 HTML(标准通用标记语言下的一个应用)文档中去执行，执行效率比完全生成 HTML 标记的 CGI 要高许多；PHP 还可以执行编译后代码，编译可以达到加密和优化代码运行的效果，使代码运行更快。

Java 是一种可以撰写跨平台应用软件的面向对象的程序设计语言。Java 技术具有卓越的通用性、高效性、平台移植性和安全性，广泛应用于个人计算机、数据中心、游戏控制台、科学超级计算机、移动电话和互联网。

C++是一个接近系统底层的综合的、支持面向对象和规范编程的程序设计语言，适用于开发要求很高的程序，如大型游戏、大型企业应用、系统应用等。

答案：C

例 26　以下关于下图所示有限自动机的叙述中，不正确的是<u>(49)</u>。(2014 年下半年试题 49)



(49) A. 该自动机识别的字符串中 a 不能连续出现

B. 自动机识别的字符串中 b 不能连续出现

C. 自动机识别的非空字符串必须以 a 结尾

D. 自动机识别的字符串可以为空串

解析：图中 a 可代表两个步骤：状态 1 -> 1，状态 2->1。如果两个 a 连续出现，则无法区分。

答案：A

例 27　由 a、b 构造且仅包含偶数个 a 的串的集合用正规式表示为<u>(49)</u>。(2009 年上半年试题 49)

(49) A. $(a^*a)^*b^*$　　B. $(b^*(ab^*a)^*)^*$　　C. $(a^*(ba^*)^*b)^*$　　D. $(a|b)^*(aa)^*$

解析：本题主要考查考生对闭包概念的理解。

Σ^* 指包括空串 ε 在内的 Σ 上所有字符串的集合。关键在于 Σ^* 可取空串 ε，理解了这个概念就不难看出答案 B 是正确的。

答案：B

例 28　对于大多数通用程序设计语言，用<u>(50)</u>描述其语法即可。(2014 年下半年试题 50)

(50) A. 正规文法　　B. 上下文无关文法

C. 上下文有关文法　　D. 短语结构文法

解析：上下文无关文法：形式语言理论中一种重要的变换文法，用来描述上下文无关语言，在乔姆斯基分层中称为 2 型文法，由于程序设计语言的语法基本上都是上下文无关

文法，因此应用十分广泛。

答案：B

例29　大多数程序设计语言的语法规则用 (49) 描述即可。(2014年上半年试题49)

(49) A．正规文法　　B．上下文无关文法

C．上下文有关文法　　D．短语结构文法

解析：形式语言理论中一种重要的变换文法，用来描述上下文无关语言，在乔姆斯基分层中称为2型文法。由于程序设计语言的语法基本上都是上下文无关文法，因此应用十分广泛。上下文无关文法拥有足够强的表达力来表示大多数程序设计语言的语法。另外，上下文无关文法又足够简单，使得我们可以构造有效的分析算法来检验一个给定字串是否是由某个上下文无关文法产生的。

答案：B

2.2.3　同步练习

1.　设某上下文无关文法如下：$S \to 11 \mid 1001 \mid S_0 \mid SS$，则该文法产生的所有二进制字符串都具有的特点是_____。

A．能被 3 整除　　B．0、1 出现的次数相等

C．0 和 1 的出现次数都为偶数　　D．能被 2 整除

2.　编译器对高级语言源程序的处理过程可以划分为词法分析、语法分析、语义分析、中间代码生成、代码优化、目标代码生成等几个阶段，其中，_____并不是每种编译器都必需的。

A．词法分析和语法分析　　B．语义分析和中间代码生成

C．中间代码生成和代码优化　　D．代码优化和目标代码生成

3.　已知某文法 $G[S]$:$S \to 0S_0$ $S \to 1$，从 S 推导出的符号串可用_____($n \geq 0$)描述。

A．$(010)^n$　　B．0^n10^n　　C．1^n　　D．01^n0

4.　有限自动机(FA)可用于识别高级语言源程序中的记号(单词)，FA 可分为确定的有限自动机(DFA)和不确定的有限自动机(NFA)。若某 DFA D 与某 NFA M 等价，则_____。

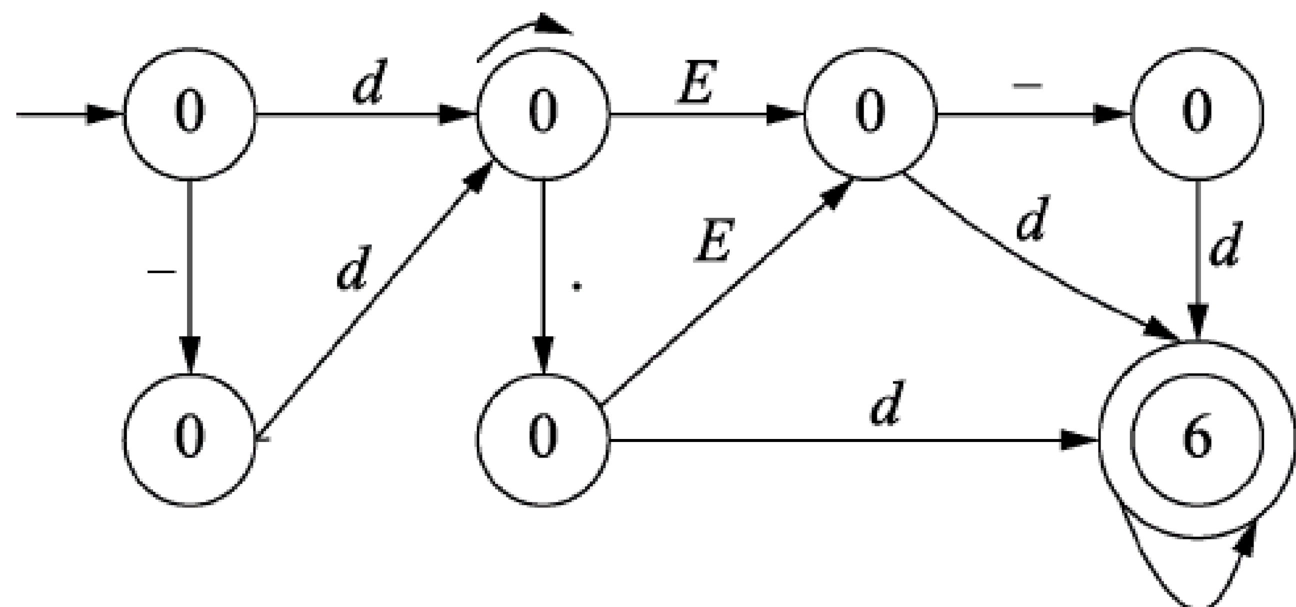
A．DFA D 与 NFA M 的状态数一定相等

B．DFA D 与 NFA M 可识别的记号相同

C．NFA M 能识别的正规集是 DFA D 所能识别的正规集的真子集

D．DFA D 能识别的正规集是 NFA M 所能识别的正规集的真子集

5.　某确定性有限自动机(DFA)的状态转换如下图所示，令 d=0|1|2|…|9，则以下字符串中，能被该 DFA 接收的是_____。



A．3857　　B．1.2E+5　　C．−123.67　　D．0.576E10

6．属于面向对象、解释型程序设计语言的是_____。(2014年下半年试题18)

A．XML　　B．Python　　C．Prolog　　D．C++

7．集合 $L=\{a^m b^m \mid m \geqslant 0\}$ _____。

A．可用正规式“a^*b^*”表示

B．不能用正规式表示，但可用非确定的有限自动机识别

C．可用正规式“$a^m b^m$”表示

D．不能用正规式表示，但可用上下文无关文法表示

8．编译程序对高级语言源程序进行翻译时，需要在该程序的地址空间中为变量指定地址，这种地址称为_____。

A．逻辑地址　　B．物理地址　　C．接口地址　　D．线性地址

9．_____是指在运行时把过程调用和响应调用所需要执行的代码加以结合。

A．绑定　　B．静态绑定　　C．动态绑定　　D．继承

10．给定文法 $G[S]$ 及其非终结符 A，FIRST(A)定义为：从 A 出发能推导出的终结符号的集合(S 是文法的起始符号，为非终结符)。对于文法 $G[S]$：

$S \rightarrow [L] \mid a$

$L \rightarrow L，S \mid S$

其中，$G[S]$包含的4个终结符号分别为：a，[ ]

则 FIRST(S)的成员包括_____。

A．a　　B．a、[　　C．a、[和]　　D．a、[、]和，

11．高级语言源程序的编译过程分若干个阶段，分配寄存器属于_____阶段的工作。

A．词法分析　　B．语法分析　　C．语义分析　　D．代码生成

2.2.4 同步练习参考答案

1．A	2．C	3．B	4．B
5．C	6．B	7．D	8．A
9．C	10．B	11．D	

2.3 本 章 小 结

本章知识点在2013年的新大纲中改动不大，只是有一些描述方面的调整。

本章主要要求考生掌握汇编、编译和解释系统的基础知识和基本工作原理，程序设计语言的基本成分，以及各类程序设计语言的主要特点和适用情况。

分析历年的考题可以看出，程序语言基础试题是每年必考的知识点，每年只考3道或者4道题目。

虽然占的比分不高，但还是需要考生掌握相应的基础知识。考查的重点是形式语言基础、语法和词法分析方法、程序控制结构。复习中还应注意补充程序语言的最新发展方向的知识。

2.4　达标训练题及参考答案

2.4.1　达标训练题

1．文法 $G=(V_T,V_N,P,S)$ 的类型由 G 中的(1)决定。若 G_0=({a,b}，{S,X,Y},P,S)，P 中的产生式及其序号如下。

① $S \to XaaY$

② $X \to YY \mid b$

③ $Y \to XbX \mid a$

则 G_0 为(2)型文法，对应于(3)，由 G_0 推导出句子 *aaaa* 和 *baabbb* 时，所用产生式序号组成的序列分别为(4)和(5)。

(1)　A．V_T　　B．V_N　　C．P　　D．S

(2)　A．0　　B．1　　C．2　　D．3

(3)　A．图灵机　　B．下推自动机　　C．有限状态自动机　　D．其他自动机

(4)～(5)　A．13133　　B．12312　　C．12322　　D．12333

2．　编译的优化工作对于下面程序段构造的控制流程图有一个基本块。

```
        A:=0
        j:=100
        i:=1
loop1:  B:=j+1
        C:=B+i
        A:=A+C
        if i=100 goto loop2
        i:=i+1
        goto loop1
loop2:  write A
        halt
```

A．1　　B．2　　C．3　　D．4

2.4.2　参考答案

1．(1) C　(2) C　(3) B　(4) D　(5) C

2．D

第 3 章　数 据 结 构

大纲要求：

- 数组(静态数组、动态数组)、线性表、链表(单向链表、双向链表、循环链表)、队列、栈、树(二叉树、查找树、平衡树、线索树、堆)、图的定义、存储和操作。
- Hash(存储地址计算、冲突处理)。

3.1　线 性 结 构

3.1.1　考点辅导

3.1.1.1　线性表

1. 线性表的定义

线性表是 n 个元素的有限序列，通常记为(a_1, a_2, …, a_n)。其特点如下。

- 存在唯一的一个称为“第一个”的元素。
- 存在唯一的一个称为“最后一个”的元素。
- 除了表头外，表中的每一个元素均只有唯一的直接前驱。
- 除了表尾外，表中的每一个元素均只有唯一的直接后继。

2. 线性表的存储结构

1)　顺序存储

线性表的顺序存储是用一组地址连续的存储单元依次存储线性表中的数据元素，从而使得逻辑关系相邻的两个元素在物理位置上也相邻。在这种存储方式下，存储逻辑关系无须占用额外的存储空间。其优点是可以随机存取表中的元素，缺点是插入和删除操作需要移动大量的元素。

一般地，在线性表的顺序存储结构中，第 i 个元素 a_i 的存储位置为

$$\mathrm{LOC}(a_i) = \mathrm{LOC}(a_1)+(i-1)\times L$$

式中，$\mathrm{LOC}(a_1)$为表中第一个元素的存储位置；L 为表中每个元素所占空间的大小。

2)　链式存储

线性表的链式存储是指用节点来存储数据元素，节点的空间可以是连续的，也可以是不连续的，因此存储数据元素的同时必须存储元素之间的逻辑关系。节点空间只有在需要的时候才申请，无须事先分配。最基本的节点结构如图 3-1 所示。

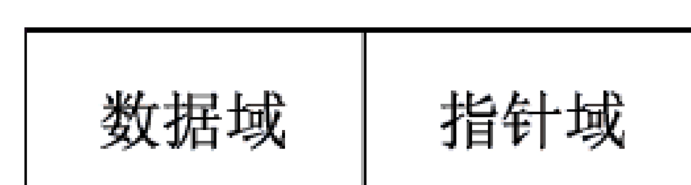


图 3-1　最基本的节点结构

其中，数据域用于存储数据元素的值，指针域则存储当前元素的直接前驱或直接后继信息，指针域中的信息称为指针(链)。n 个节点通过指针连成一个链表，若节点中只有一个指针域，则称为线性链表(单链表)。

线性表采用链表作为存储结构时，不能进行数据元素的随机访问，但其优点是插入和删除操作不需要移动元素。以下是几种其他链表结构。

(1) 双向链表。每个节点包含两个指针，指明直接前驱和直接后继元素，可在两个方向上遍历链表。

(2) 循环链表。表尾节点的指针指向表中的第一个节点，可在任何位置上开始遍历整个链表。

(3) 静态链表。借助数组来描述线性表的链式存储结构。

在链式存储结构中，只需要一个指针(头指针)指向第一个节点，就可以顺序访问到表中的任意一个元素。为了简化对链表状态的判定和处理，特别引入一个不存储数据元素的节点，称为头节点，将其作为链表的第一个节点并令头指针指向该节点。

3. 线性表的插入和删除运算

1) 基于顺序存储结构的运算

插入元素前要移动元素以挪出空的存储单元，然后再插入元素；删除元素时同样需要移动元素，以填充被删除的存储单元。在等概率下平均移动元素的次数分别是

$$E_{\text{insert}}=\sum_{i=1}^{n+1}P_i\times(n-i+1)=\frac{1}{n+1}\sum_{i=1}^{n+1}(n-i+1)=\frac{n}{2}$$

$$E_{\text{delete}}=\sum_{i=1}^{n}q_i\times(n-i)=\frac{1}{n}\sum_{i=1}^{n}(n-i)=\frac{n-1}{2}$$

2) 基于链式存储结构的运算

在链式存储结构下进行插入和删除，其实质都是对相关指针的修改。

(1) 在单向链表中插入节点时，指针的变化情况如图 3-2 所示。

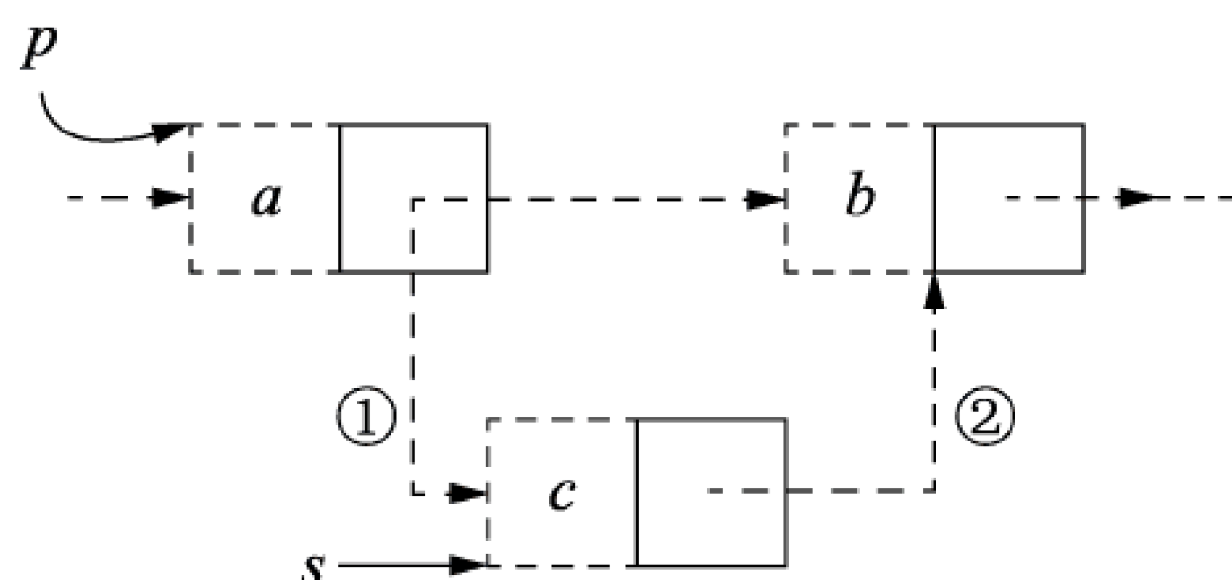


图 3-2　单向链表插入节点时的指针变化情况

(2) 在单向链表中删除节点时，指针的变化情况如图 3-3 所示。

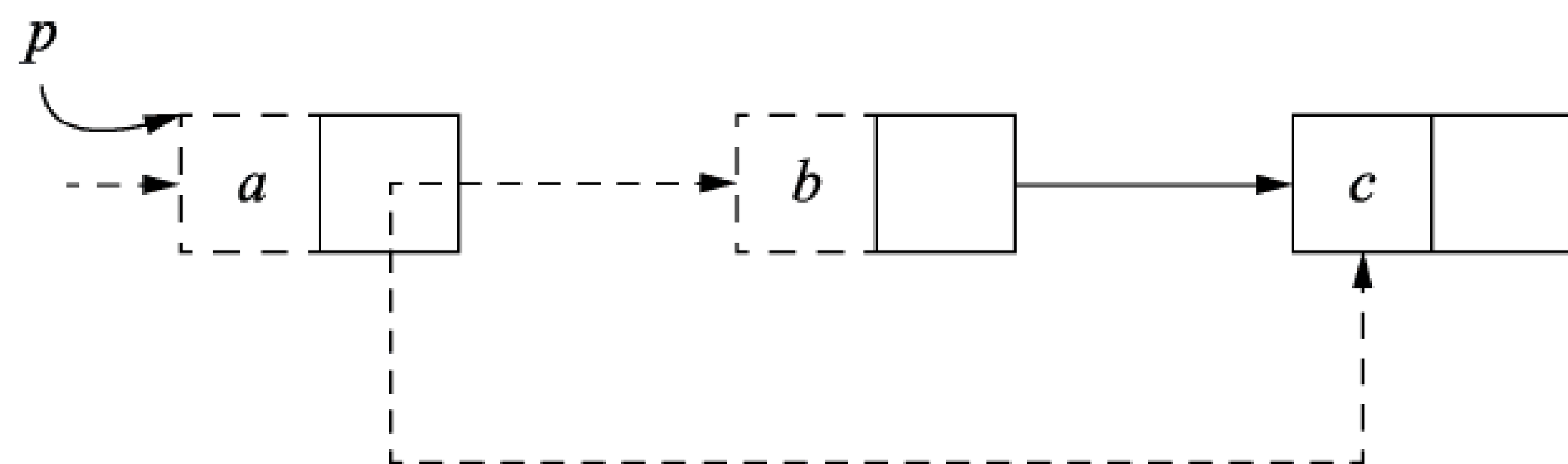


图 3-3　单向链表删除节点时的指针变化情况

(3) 在双向链表中插入节点时，指针的变化情况如图 3-4 所示。

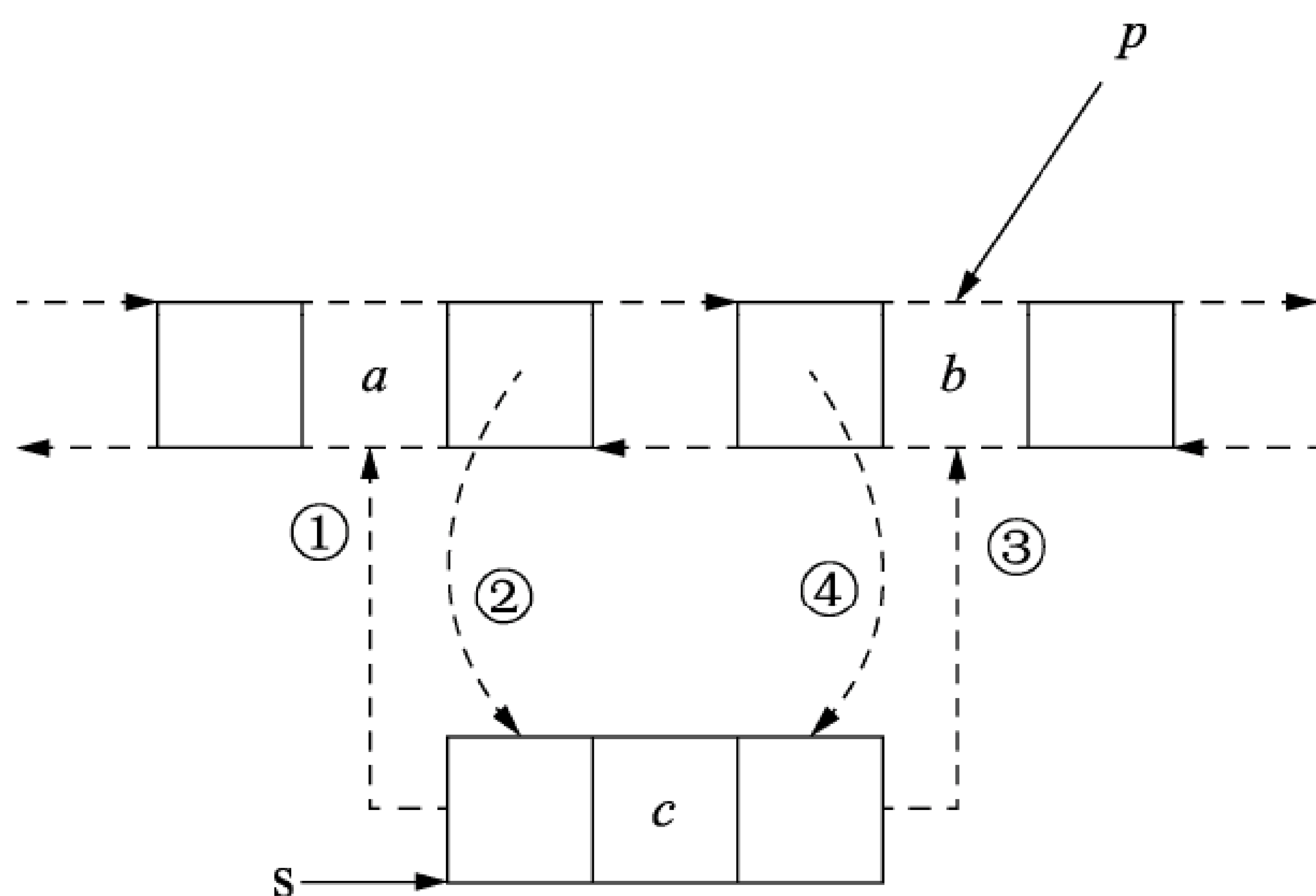


图 3-4 双向链表插入节点时的指针变化情况

(4) 在双向链表中删除节点时，指针的变化情况如图 3-5 所示。

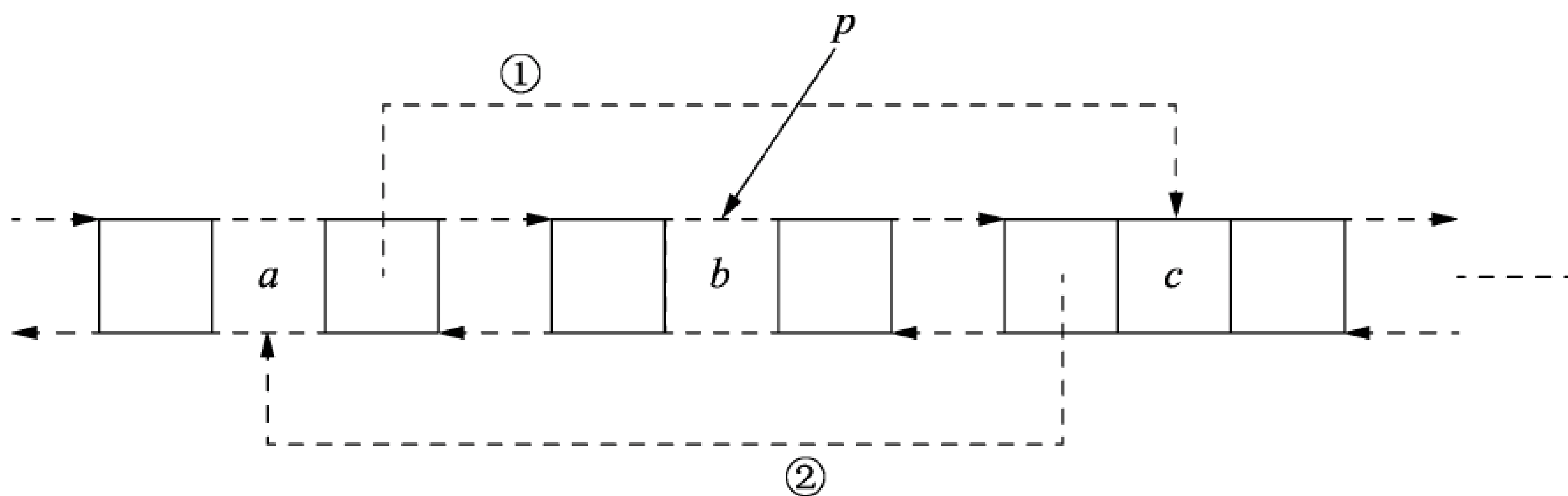


图 3-5 双向链表删除节点时的指针变化情况

注意： 图 3-2 至图 3-5 中①为插入运算前的指针走向；②为插入运算后的指针走向；虚线为插入后的指针指向。

3.1.1.2 栈和队列

1. 栈

1) 栈的定义及基本运算

栈是只能通过访问它的一端来实现数据存储和检索的一种线性数据结构。栈的修改是按先进后出的原则进行的。因此，栈又称为先进后出(FILO，或后进先出)的线性表。栈进行插入和删除操作的一端称为栈顶，另一端称为栈底。不含数据元素的栈称为空栈。

对栈进行的基本操作有以下几种。

- 置空栈 InitStack(S)：创建一个空栈 S。
- 判栈空 Empty(S)：当栈 S 为空栈时返回真值；否则返回假值。
- 入栈 Push(S, x)：将元素 x 加入栈顶，并更新栈顶指针。
- 出栈 Pop(S)：将栈顶元素从栈中删除，并更新栈顶指针。若需要得到栈顶元素的值，可将 Pop(S)定义为一个函数，它返回栈顶元素的值。
- 读栈顶元素 Top(S)：返回栈顶元素的值，但不修改栈顶指针。

2)　栈的存储结构

(1)　顺序存储。栈的顺序存储是指用一组地址连续的存储单元依次存储自栈顶到栈底的数据元素，同时附设指针 top 指示栈顶元素的位置。在顺序存储方式下，需要预先定义或申请栈的存储空间，也就是说，栈空间的容量是有限的。因此在顺序栈中，当一个元素入栈时，需要判断是否栈满，若栈满，则元素入栈会发生上溢现象。

利用栈底位置不变的特性，可以让两个顺序栈共享一个一维数据空间，以互补余缺，实现方法是：将两个栈的栈底位置分别设在存储空间的两端，让它们的栈顶各自向中间延伸。这样，两个栈的空间就可以相互调节，只有在整个存储空间被占满时才发生上溢，这样一来产生上溢的概率要小得多。

(2)　链式存储。用链表作为存储结构的栈也称为链栈。由于栈中元素的插入和删除仅在栈顶一端进行，因此不必设置头节点，链表的头指针就是栈顶指针。

3)　栈的应用

栈的典型应用包括表达式求值、括号匹配等，在计算机语言的实现中以及将递归过程转变为非递归过程的处理中，栈有重要的作用。

2.　队列

1)　队列的定义及基本运算

队列是一种先进先出(FIFO)的线性表，它只允许在表的一端插入元素，而在表的另一端删除元素。在队列中，允许插入元素的一端称为队尾(Rear)，允许删除元素的一端称为队头(Front)。

对队列进行的基本操作如下。

(1)　置队空 InitQueue(Q)：创建一个空的队列 Q。

(2)　判队空 Empty(Q)：判断队列是否为空。

(3)　入队 EnQueue(Q, x)：将元素 x 加入到队列 Q 的队尾，并更新队尾指针。

(4)　出队 DeQueue(Q)：将队头元素从队列 Q 中删除，并更新队头指针。

(5)　读队头元素 Frontque(Q)：返回队头元素的值，但并不更新队头指针。

2)　队列的存储结构

(1)　顺序存储。队列的顺序存储结构是利用一组地址连续的存储单元存放队列中的元素。由于队列中元素的插入和删除限定在队列的两端进行，因此设置队头指针和队尾指针，分别指示当前的队首元素和队尾元素。

在顺序队列中，为了降低运算的复杂度，元素入队时只需修改队尾指针，元素出队时只需修改队头指针。由于顺序队列的存储空间是提前设定的，所以队尾指针会有一个上限值，当队尾指针达到其上限时，就不能只通过修改队尾指针来实现新元素的入队操作了。此时，可通过整除取余运算将顺序队列假想成一个环状结构，称之为循环队列。在队列空和队列满的情况下，循环队列的队头、队尾指针指向的位置是相同的。为了区别队空和队满的情况，可采用两种处理方式：其一是设置一个标志位，以区别头、尾指针的值相同时队列是空还是满；其二是牺牲一个元素空间，约定以“队列的尾指针所指位置的下一个位置是头指针时”表示队列满，头、尾指针的值相同时表示队列空。

(2)　链式存储。用链表表示的队列简称为链队列。为了便于操作，给链队列添加一个头节点，并令头指针指向头节点。队列为空的判定条件是：头指针和尾指针的值相同，且均指向头节点。

3)　队列的应用

队列结构常用于处理需要排队的场合，如操作系统中处理打印任务的打印队列、离散事件的计算机模拟等。

3.1.1.3　串

1. 串的定义及基本运算

串是仅由字符构成的有限序列，是取值范围受限的线性表。一般记为 S=‘$a_1a_2\cdots a_n$’，其中 S 是串名，$a_1\,a_2\cdots a_n$ 是串值。

下面介绍串的几个基本概念。

(1)　空串：长度为零的串，空串不包含任何字符。

(2)　空格串：由一个或多个空格组成的串。

(3)　子串：由串中任意长度的连续字符构成的序列。含有子串的串称为主串。子串在主串中的位置指子串首次出现时，该子串的第一个字符在主串中的位置。空串是任意串的子串。

(4)　串相等：指两个串长度相等且对应位置上的字符也相同。

(5)　串比较：两个串比较大小时以字符的 ASCII 码值作为依据。比较操作从两个串的第一个字符开始进行，字符的 ASCII 码值大者所在的串为大；若其中一个串先结束，则以串长较大者为大。

对串进行的基本操作有以下几种。

(1)　赋值操作 StrAssign(s, t)：将串 t 的值赋给串 s。

(2)　连接操作 Concat(s, t)：将串 t 接续在串 s 的尾部，形成一个新串。

(3)　求串长 StrLength(s)：返回串 s 的长度。

(4)　串比较 StrCompare(s, t)：比较两个串的大小。

(5)　求子串 SubString(s, start, len)：返回串 s 中从 start 开始的、长度为 len 的字符序列。

2. 串的存储结构

1)　串的静态存储：定长存储结构

串的顺序存储结构是用一组地址连续的存储单元来存储串值的字符序列。由于串中的元素为字符，所以可通过程序语言提供的字符数组定义串的存储空间，也可以根据串长的需要动态申请字符串的空间。

2)　串的链式存储：块链

串也可采用链表方式作为存储结构，当用链表存储串中的字符时，每个节点中可以存储一个字符，也可以存储多个字符，要考虑存储密度的问题。在链式存储结构中，节点大小的选择和顺序存储方法中数组空间大小的选择一样重要，它直接影响对串处理的效率。

3. 串的模式匹配

子串的定位操作通常称为串的模式匹配，它是各种串处理系统中最重要的运算之一。子串也称为模式串。

1)　朴素的模式匹配算法

朴素的模式匹配算法也称为布鲁特—福斯算法，其基本思想是：从主串的第一个字符起与模式串的第一个字符比较，若相等则继续逐个字符进行后续的比较，否则从主串的第

二个字符起与模式串的第一个字符重新比较，直至模式串中的每个字符依次和主串中的一个连续的字符序列相等，则称匹配成功，否则称匹配失败。

该算法在最好情况下匹配算法的时间复杂度为 $O(n+m)$，而在最坏情况下的时间复杂度为 $O(n\times m)$。

2)　改进的模式匹配算法

改进的模式匹配算法又称为 KMP 算法，其改进之处在于：每当匹配过程中出现相比较的字符不相等时，不需要回溯主串的指针，而是利用已经得到的“部分匹配”的结果，将模式串向后“滑动”尽可能远的距离，再继续进行比较。

此算法的时间复杂度为 $O(n+m)$。

3.1.2　典型例题分析

例 1　采用顺序表和单链表存储长度为 n 的线性序列，根据序号查找元素，其时间复杂度分别为 (51) 。(2013 年上半年试题 51)

(51) A. O(1)、O(1)　　B. O(1)、O(n)

C. O(n)、O(1)　　D. O(n)、O(n)

解析：顺序表存储位置是相邻连续的，可以随机访问的一种数据结构，一个顺序表在使用前必须指定起始长度，一旦分配内存，则在使用中不可以动态更改。它的优点是访问数据比较方便，可以随机访问表中的任何一个数据。链表是通过指针来描述元素关系的一种数据结构，它可以是物理地址不连续的物理空间。不能随机访问链表元素，必须从表头开始，一步一步搜索元素。它的优点是：对于数组，可以动态地改变数据的长度，分配物理空间。因此两者的查找复杂度就显而易见了。

答案：B

例 2　若元素以 a、b、c、d、e 的顺序进入一个初始为空的栈中，每个元素进栈、出栈各 1 次，要求出栈的第一个元素为 d，则合法的出栈序列共有 (57) 种。(2016 年上半年试题 57)

(57) A. 4　　B. 5　　C. 6　　D. 24

解析：一共 5 个元素 a、b、c、d、e，而 d 被要求作为第一个元素出栈。当 d 出栈后的情况应为：

有一个元素 e 还未入栈，而栈中已有 a、b、c。栈中的 a、b、c 出栈顺序是已无可变性，必须是 c、b、a，此时，只是分析 e 在什么位置出栈即可。

c、b、a 3 个元素，有 4 个空位，所以可以产生的序列可能为：

(1)　d，e，c，b，a

(2)　d，c，e，b，a

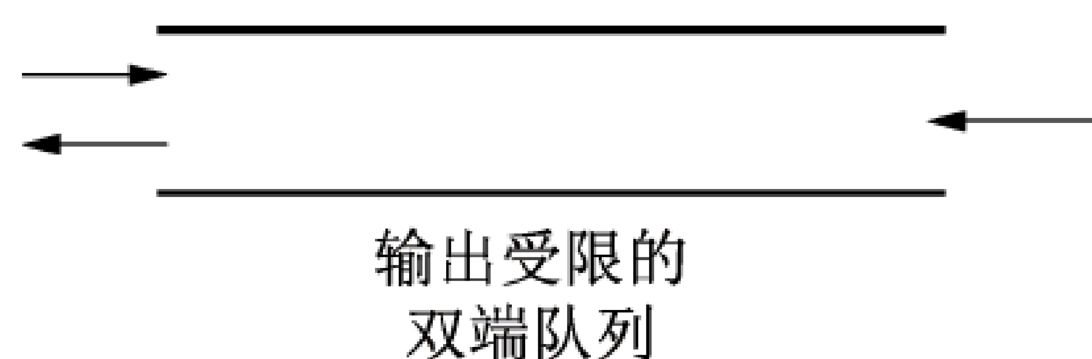
(3)　d，c，b，e，a

(4)　d，c，b，a，e

答案：A

例 3　输出受限的双端队列是指元素可以从队列的两端输入、但只能从队列的一端输

出，如下图所示。若有 e1、e2、e3、e4 依次进入输出受限的双端队列，则得不到输出队列 (53) 。(2013 年上半年试题 53)



(53) A. e4、e3、e2、e1　　B. e4、e2、e1、e3
C. e4、e3、e1、e2　　D. e4、e2、e3、e1

解析：此题考查队列的性质，队列为先进先出的线性结构，题中给出的受限的双端队列，两端都可以进，而一端可出，假设分 a 和 b 端，b 端可以进出，由 D 选项的出序列，可以看出 e1、e2、e3 按顺序从 a 端进入，而 e4 从 b 端进入，当 e4 从 b 端出来之后，无法将后面的 e2 出队列，故 D 选项有误。

答案：D

例 4　对于一个长度大于 1 且不存在重复元素的序列，令其所有元素依次通过一个初始为空的队列后，再通过一个初始为空的栈。设队列和栈的容量都足够大，一个序列通过队列(栈)的含义是序列的每个元素都入队列(栈)且出队列(栈)一次且仅一次。对于该序列在上述队列和栈上的操作，正确的是 (57) 。(2012 年上半年试题 57)

(57) A. 出队序列和出栈序列一定相同
B. 出队序列和出栈序列一定互为逆序
C. 入队序列和出队序列一定相同，入栈序列和出栈序列不一定相同
D. 入栈序列和出栈序列一定互为逆序，入队序列和出队序列不一定互为逆序

解析：队列具有先进先出的特点，也就是说，最先入队的元素最先出队，所以入队序列和出队序列一定相同。栈则具有先进后出的特点，如果所有元素进栈后再依次出栈，则入栈序列和出栈序列互为逆序；否则不一定。

答案：C

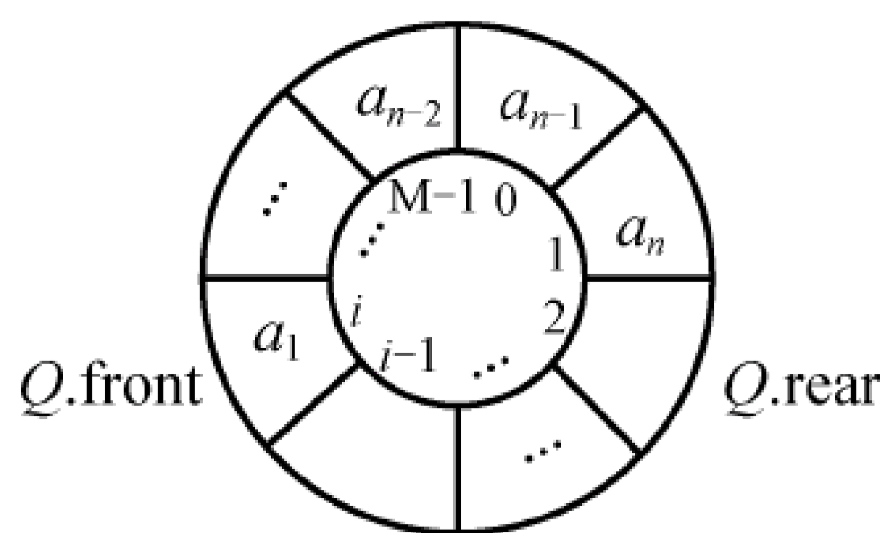
例 5　对于线性表(由 *n* 个同类元素构成的线性序列)，采用单向循环链表存储的特点之一是 (58) 。(2011 年下半年试题 58)

(58) A. 从表中任意节点出发都能遍历整个链表
B. 对表中的任意节点可以进行随机访问
C. 对于表中的任意一个节点，访问其直接前驱和直接后继节点所有时间相同
D. 第一个节点必须是头节点

解析：对于单向循环链表，可以从表中任意节点出发都能遍历整个链表。但并不能对表中的任意节点进行随机访问，需要从设置的第一个节点开始，沿着指针访问表中的节点。当然访问某一节点的直接后继节点最快，访问其直接前驱节点最慢，因为首先要遍历到表尾，然后从表头遍历到其前驱节点。

答案：A

例 6　设某循环队列 Q 的定义中有 front 和 rear 两个域变量，其中，front 指示队头元素的位置，rear 指示队尾元素之后的位置，如下图所示。若该队列的容量为 M，则其长度为 (57) 。(2013 年下半年试题 57)



(57) A. (Q.rear-Q.front+1)　　　　B. (Q.rcar-Q.front+M)

C. (Q.rear-Q.front+1)%M　　　　D. (Q.rear-Q.front+M)%M

解析：初始时，队列为空，Q.rear=Q.front=0；随着不断入队、出队操作，Q.rear 和 Q.front 的值不断变化，出队时 Q.front=(Q.front+1)%M，入队时 Q.rear=(Q.rear+1)%M。如果 Q.rear>Q.front，则队列长度为 Q.rear-Q.front=(Q.rear-Q.front+M)%M；如果 Q.rear>Q.front，则队列长度为(Q.rear-Q.front+M)%M。因此，队列长度应为(Q.rear-Q.front+M)%M。

答案：D

例 7　以下关于线性表存储结构的叙述，正确的是__(57)__。(2013 年下半年试题 57)

(57) A. 线性表采用顺序存储结构时，访问表中任意一个指定序号元素的时间复杂度为常量级

B. 线性表采用顺序存储结构时，在表中任意位置插入新元素的运算时间复杂度为常量级

C. 线性表采用链式存储结构时，访问表中任意一个指定序号元素的时间复杂度为常量级

D. 线性表采用链式存储结构时，在表中任意位置插入新元素的运算时间复杂度为常量级

解析：顺序存储结构可以随机存取，时间复杂度最低为常量级的，答案选 A。

答案：A

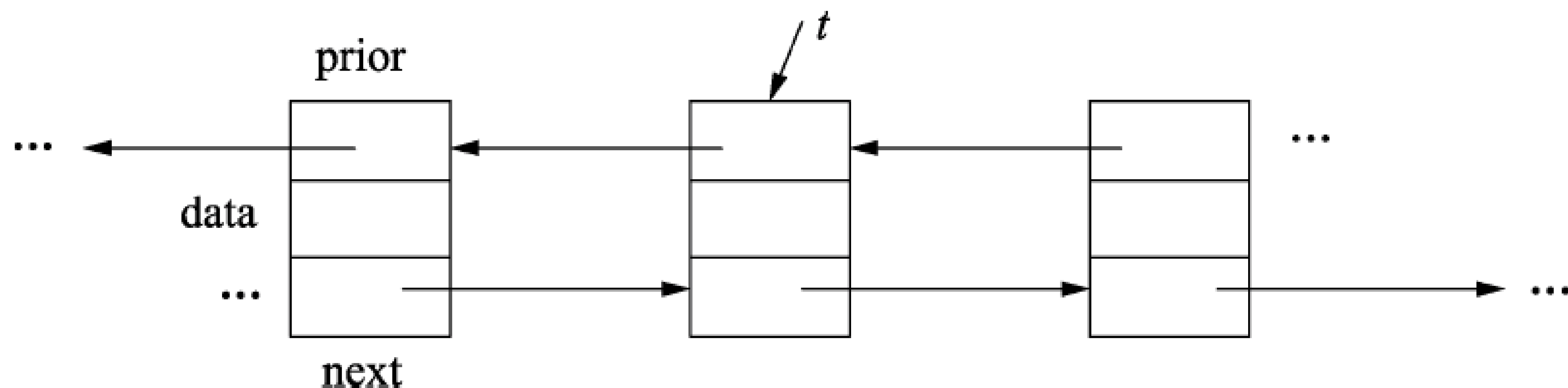
例 8　某双向链表中的节点如下图所示，删除 t 所指节点的操作为__(54)__。(2009 年下半年试题 54)

(54) A. t->prior->next = t->next；t->next->prior = t->prior；

B. t->prior->prior = t->prior；t->next->next = t->next；

C. t->prior->next = t->prior；t->next->prior = t->next；

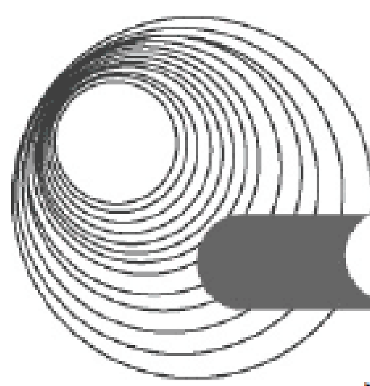
D. t->prior->prior = t->next；t->next->prior = t->prior；



解析：本题考查双向链表的基本操作。

双向链表每个数据节点中都有两个指针，分别指向直接后继和直接前驱。所以，从双向链表中的任意一个节点开始，都可以很方便地访问它的前驱节点和后继节点。

删除 t 节点，只需把 t 原来的前驱的 next 指向 t 现在的后继，t 原来后继的 prior 指向 t



现在的前驱即可。

答案：A

例 9 单向链表中往往含有一个头节点，该节点不存储数据元素，一般令链表的头指针指向该节点，而该节点指针域的值为第一个元素节点的指针。以下关于单链表头节点的叙述中，错误的是__(60)__。(2009 年下半年试题 60)

(60) A．若在头节点中存入链表长度值，则求链表长度运算的时间复杂度为 $O(1)$

B．在链表的任何一个元素前后进行插入和删除操作可用一致的方式进行处理

C．加入头节点后，代表链表的头指针不因为链表为空而改变

D．加入头节点后，在链表中进行查找运算的时间复杂度为 $O(1)$

解析：本题考查单链表头节点的相关知识。

A 选项：由于在头节点中存入链表长度值，在遍历链表时先从头指针开始，头指针指向头节点，头节点的数据域即为链表长度，故 A 正确。

B 选项：①插入运算是将值为 x 的新节点插入到表的第 i 个节点的位置上，即插入到 a_{i-1} 与 a_i 之间。因此，必须首先找到 a_{i-1} 的存储位置 p，然后生成一个数据域为 x 的新节点，并令 q 指针指向该新节点，新节点的指针域指向节点 a_i。从而实现 3 个节点 a_{i-1}、x 和 a_i 之间的逻辑关系的变化。

定位 a_{i-1} 并将指针 p 指向它，代码如下。

```
q = new LNode;
q->data=x;
q->next=p->next;
p->next=q;
```

② 删除运算是将表的第 i 个节点删去。因为在单链表中节点 a_i 的存储地址是在其直接前驱节点 a_{i-1} 的指针域 next 中，所以必须首先找到 a_{i-1} 的存储位置 p。然后令 p→next 指向 a_i 的直接后继节点，即把 a_i 从链上摘下。最后释放节点 a_i 的空间。

代码如下。

```
r=p->next;
p->next=r->next;
delete r;
```

故 B 正确。

C 选项：头节点引入，即增加一个表头节点，数据域可根据需要使用或不用。特点如下。

- 表中第一个节点和在表的其他位置上的操作一致，无须进行特殊处理。
- 无论链表是否为空，其头指针是指向头节点。因此空表和非空表的处理统一。

故 C 正确。

D 选项：查找过程从开始节点出发，顺着链表逐个将节点的值和给定值 key 作比较。算法如下。

```
LNode *locatenode(head, key)
 {
LNode *p;
```

```
p=head->next;
  while( p && p->data!=key)
     p=p->next;
   return p;
}
```

该算法的执行时间也与输入实例中的取值 key 有关，其平均时间复杂度的分析类似于按序号查找。故 D 错误。

答案：D

例 10　对于一个长度为 n(n>1)且元素互异的序列，将其所有元素依次通过一个初始为空的栈后，再通过一个初始为空的队列。假设队列和栈的容量都足够大，且只要栈非空就可以进行出栈操作，只要队列非空就可以进行出队操作，那么以下叙述中正确的是<u> (57) </u>。(2015 上半年试题 57)

(57) A．出队序列和出栈序列一定互为逆序

B．出队序列和出栈序列一定相同

C．入栈序列与入队序列一定相同

D．入栈序列与入队序列一定互为逆序

解析：由题干可知，出栈之后，直接入队，然后出队。而入队序列等于出栈序列，出队序列等于入队序列，所以出队序列和出栈序列一定相同。

答案：B

例 11　字符串采用链表存储方式时，每个节点存储多个字符有助于提高存储密度。若采用节点大小相同的链表存储串，则串比较、求子串、串连接、串替换等串的基本运算中，<u>(62)</u>。(2009 年下半年试题 62)

(62) A．进行串的比较运算最不方便　　B．进行求子串运算最不方便

C．进行串连接最不方便　　D．进行串替换最不方便

解析：在用链表作为字符串的存储方式时，如果每个节点存储多个字符，进行串连接、串替换和串的比较等操作时，不会有很大影响，但是在进行子串求解时，因为有可能涉及所求的子串不在一个节点上存储，所以会比较麻烦，因此总的来说，进行求子串时最不方便。

答案：B

例 12　设栈 S 和队列 Q 的初始状态为空，元素 a、b、c、d、e、f 依次进入栈 S。要求每个元素出栈后立即进入队列 Q，若 7 个元素出队列的顺序为 b、d、f、e、c、a、g，则栈 S 的容量最小应该是<u> (58) </u>。(2015 年下半年试题 58)

(58) A. 5　　B. 4　　C. 3　　D. 2

解析：由于队列先进先出，所以出队的顺序即为入队的顺序，也就是元素出栈的顺序。a、b 依次进栈，b 最先出栈，a 暂时没出栈；然后 c、d 进栈，d 出栈，c 仍在栈里，此时栈里有 a、c；接着 e、f 入栈，f 出栈、e 出栈、c 出栈、a 出栈；最后 g 入栈并出栈。可见，f 入栈后，栈中的元素最多，有 a、c、e、f 这 4 个元素，所以栈的最小容量为 4。

答案：B

3.1.3 同步练习

1. 设栈S和队列Q的初始状态为空，元素按照a、b、c、d、e的次序进入栈S，当一个元素从栈中出来后立即进入队列Q。若队列的输出元素序列是c、d、b、a、e，则元素的出栈顺序是__(1)__，栈S的容量至少为__(2)__。

(1) A. a、b、c、d、e　　B. e、d、c、b、a
　　C. c、d、b、a、e　　D. e、a、b、d、c

(2) A. 2　　B. 3　　C. 4　　D. 5

2. 对于$n(n \geq 0)$个元素构成的线性序列L，在______时适合采用链式存储结构。

A. 需要频繁修改L中元素的值
B. 需要频繁地对L进行随机查找
C. 需要频繁地对L进行删除和插入操作
D. 要求L存储密度高

3.1.4 同步练习参考答案

1. (1) C　　(2) B
2. C

3.2 数组、矩阵和广义表

3.2.1 考点辅导

3.2.1.1 数组

1. *数组的定义及基本运算*

n维数组是一种“同构”的数据结构，其每个元素类型相同、结构一致。数组是定长线性表在维数上的扩张，即线性表中的元素又是一个线性表。

数组结构的特点是：数据元素数目固定；数据元素具有相同的类型；数据元素的下标关系具有上下界的约束且下标有序。

对数组进行的基本运算有以下两种。

(1) 给定一组下标，存取相应的数据元素。

(2) 给定一组下标，修改相应的数据元素中某个数据项的值。

2. *数组的顺序存储*

一旦定义了数组，结构中的数据元素个数和元素之间的关系就不再发生变动，因此数组适合于采用顺序存储结构。

由于计算机的内存结构是一维线性的，因此存储多维数组时必须按照某种方式进行降维处理，即将数组元素排成一个线性序列，这就产生了次序约定问题。对二维数组有两种存储方式：一种是以列为主序的存储方式；另一种是以行为主序的存储方式。

设每个数据元素占用 L 个单元，m、n 为数组的行数和列数，那么以行为主序优先存储的地址计算公式为

$$\mathrm{Loc}(a_{ij})=\mathrm{Loc}(a_{11})+((i-1)n+(j-1))L$$

同样的，以列为主序优先存储的地址计算公式为

$$\mathrm{Loc}(a_{ij})=\mathrm{Loc}(a_{11})+((j-1)m+(i-1))L$$

3.2.1.2　矩阵

1. 特殊矩阵

若矩阵中元素(或非零元素)的分布有一定的规律，则称之为特殊矩阵。常见的特殊矩阵有对称矩阵、三角矩阵、对角矩阵等。

对称矩阵：若矩阵 $\boldsymbol{A}_{n\times n}$ 中的元素有

$$a_{ij}=a_{ji}\quad 1\leqslant i，j\leqslant n$$

则称之为 n 阶对称矩阵。

上(下)三角矩阵：矩阵的上(下)三角(不包括对角线)中的元素均为常数或零。

对角矩阵：矩阵中的非零元素都集中在以主对角线为中心的带状区域中，即除了主对角线上和在对角线上、下方若干条对角线上的元素外，其余的矩阵元素都为零。

2. 稀疏矩阵

在一个矩阵中，若非零元素的个数远远少于零元素的个数，且非零元素的分布没有规律，则称之为稀疏矩阵。存储稀疏矩阵的非零元素时必须同时存储其位置(行、列号)，用三元组(i，j，a_{ij})可唯一确定矩阵中的一个元素。因此，一个稀疏矩阵可由表示非零元素的三元组及其行、列数唯一确定。

稀疏矩阵的三元组表的顺序存储结构称为三元组顺序表，常用的三元组表的链式存储结构是十字链表。

3.2.1.3　广义表

1. 广义表的定义

广义表是线性表的推广，是由零个或多个单元素或子表所组成的有限序列。

广义表与线性表的区别在于：线性表的元素都是结构上不可分的单元素，而广义表的元素既可以是单元素也可以是有结构的表。

广义表一般记为

$$\mathrm{LS}=(\alpha_1,\alpha_2,\cdots,\alpha_n)$$

式中，$\alpha_i(1\leqslant i\leqslant n)$ 既可以是单个元素，又可以是广义表，分别称为原子和子表。

广义表的长度是指广义表中元素的个数；深度是指广义表展开后所含括号的最大层数。

2. 广义表的基本操作

下面介绍广义表的基本操作。

(1) 取表头 head(LS)。非空广义表 LS 的第一个元素称为表头，它可以是一个单元素，也可以是一个子表。

(2) 取表尾 tail(LS)。在非空广义表中，除表头元素之外，由其余元素所构成的表称为表尾。非空广义表的表尾必定是一个表。

3. 广义表的特点

广义表具有以下特点。

(1) 广义表可以是多层次的结构，因为广义表的元素可以是子表，而子表的元素还可以是子表。

(2) 广义表中的元素可以是已经定义的广义表的名字，所以一个广义表可被其他广义表所共享。

(3) 广义表可以是一个递归的表，即广义表中的元素也可以是本广义表的名字。

4. 广义表的存储结构

广义表通常采用链式存储结构。若广义表不空，则可分解为表头和表尾两部分；反之，一对确定的表头和表尾可唯一决定一个广义表。

3.2.2 典型例题分析

例 1 设下三角矩阵(上三角部分的元素值都为 0)$A[0..n,\ 0..n]$如下所示，将该三角矩阵的所有非零元素(即行下标不小于列下标的元素)按行优先压缩存储在容量足够大的数组$M[1..m]$中，则元素 $A[i,j](0\leqslant i\leqslant n, j\leqslant i)$存储在数组 M 的__(57)__中。(2011 年上半年试题 57)

$$\begin{bmatrix} A_{00} & & & & & & \\ A_{10} & A_{11} & & & & 0 & \\ \bullet & & \bullet & & & & \\ \bullet & & & \bullet & & & \\ \bullet & & & & \bullet & & \\ A_{70} & A_{71} & A_{72} & & \cdots & A_{77} & \\ A_{80} & A_{81} & A_{82} & A_{83} & & \cdots & A_{88} \end{bmatrix}$$

(57) A. $M\left[\dfrac{i(i+1)}{2}+j+1\right]$　　B. $M\left[\dfrac{i(i+1)}{2}+j\right]$

C. $M\left[\dfrac{i(i-1)}{2}+j\right]$　　D. $M\left[\dfrac{i(i-1)}{2}+j+1\right]$

解析：第 0 行有 1 个元素保存在数组 M 中，第 1 行有两个元素保存在数组 M 中，第 i−1 行中有 i 个元素保存在数组 M 中，第 i 行之前有 $1+2+3+\cdots+i=i(i+1)/2$ 个元素保存在数组 M 中，元素 $A[i,j]$是第 i 行的 j+1 个元素。由于数组 M 的下标从 1 开始，因此 $A[i,j]$的值存储在$M\left[\dfrac{i(i+1)}{2}+j+1\right]$中。

答案：A

例 2 设某 n 阶三对角矩阵 $A_{n\times n}$ 的示意图如下图所示。若将该三对角矩阵的非零元素

按行存储在一维数组 $B[k](1 \leqslant k \leqslant 3n-2)$中，则 k 与 i、j 的对应关系是 (58) 。(2015 年上半年试题 58)

$$A_{n \times n}=\begin{bmatrix} a_{1,1} & a_{1,2} & & & & 0 \\ a_{2,1} & a_{2,2} & a_{2,3} & & & \\ & a_{3,2} & a_{3,3} & a_{3,4} & & \\ & & \cdots & \cdots & \cdots & \\ & 0 & & \cdots & \cdots & \cdots \\ & & & & a_{n,n-1} & a_{n,n} \end{bmatrix}$$

(58) A．$k=2i+j-2$　　B．$k=2i-j+2$　　C．$k=3i+j-1$　　D．$k=3i-j+2$

解析：本题中的矩阵为三对角矩阵，$k=3(i-1)-1+j-i+1+1=2i+j-2$。本题可以采用代入验证法。例如，当 $i=1$，$j=1$ 时，$k=1$。

选项 A：$k=2i+j-2=2+1-2=1$；

选项 B：$k=2i-j+2=2-1+2=3$；

选项 C：$k=3i+j-1=3+1-1=3$；

选项 D：$k=3i-j+2=3+1+2=4$。

此时可以排除 B、C、D，直接选 A。

答案：A

例 3　设 L 为广义表，将 head(L)定义为取非空广义表的第一个元素，tail(L)定义为取非空广义表除第一个元素外剩余元素构成的广义表。若广义表 $L=((x, y, z), a, (u, t, w))$，则从 L 中取出原子项 y 的运算是(62)。(2009 年上半年试题 62)

(62) A．head(tail(tail(L)))　　B．tail(head(head(L)))

C．head(tail(head(L)))　　D．tail(tail(head(L)))

解析：对于广义表 $L=((x，y，z)，a，(u，t，w))$，head(L)定义为取非空广义表的第一个元素，tail(L)定义为取非空广义表除第一个元素外剩余元素构成的广义表。head(tail(head(L)))即为先取 L 的第一个元素$(x，y，z)$；接着取除第一个元素外的剩余元素 y、z；然后取第一个元素即 y。

答案：C

3.2.3　同步练习

1.　广义表中的元素可以是原子，也可以是表，因此广义表的适用存储结构是______。

A．链表　　B．静态数组　　C．动态数组　　D．散列表

2.　若有数组声明 a[0..3，0..2，1..4]，设编译时为 a 分配的存储空间首地址为 base_a，且每个数组元素占据一个存储单元。当元素以行为序存放(即按 a[0, 0, 1]、a[0, 0, 2]、a[0, 0, 3]、a[0, 0, 4]、a[0, 1, 1]、a[0, 1, 2]、…、a[3, 2, 4]顺序存储)时，则数组元素 a[2, 2, 2]在其存储空间中相对 base_a 的偏移量是______。

A．8　　B．12　　C．33　　D．48

3.2.4　同步练习参考答案

1．A　2．C

3.3　树

3.3.1　考点辅导

3.3.1.1　树的定义及基本运算

树是 $n(n\geqslant0)$个节点的有限集合，n=0 时称为空树，在任一非空树中：

(1)　有且仅有一个称为根的节点。

(2)　其余的节点可分为 $m(m\geqslant0)$个互不相交的子集 $T_1, T_2, \cdots, T_m$，其中每个子集本身又是一棵树，并称其为根节点的子树。

树的递归定义表明了树的固有特性，也就是一棵树由若干棵子树构成，而子树又由更小的子树构成。

树中的基本概念如下。

(1)　双亲和孩子。节点的子树的根称为该节点的孩子；该节点称为其子节点的双亲。

(2)　兄弟。具有相同双亲的节点互为兄弟。

(3)　节点的度。一个节点的子树的个数记为该节点的度。

(4)　叶子节点。也称为终端节点，指度为零的节点。

(5)　内部节点。度不为零的节点称为分支节点或非终端节点。除根节点之外，分支节点也称为内部节点。

(6)　节点的层次。根为第一层，根的孩子为第二层，以此类推。

(7)　树的高度。一棵树的最大层次数记为树的高度(或深度)。

(8)　有序(无序)树。若将树中的节点的各子树看成是从左到右具有次序的，即不能交换，则称该树为有序树；否则称为无序树。

(9)　森林。是 $m(m\geqslant0)$棵互不相交的树的集合。

3.3.1.2　二叉树

1．二叉树的定义

二叉树(Binary Tree)是 $n(n\geqslant0)$个节点的有限集合，它或者是空树(n=0)，或者是由一个根节点及两棵互不相交的、分别称为左子树和右子树的二叉树所组成。

二叉树与树的区别如下。

- 二叉树的节点的子树要区分左子树和右子树，即使在节点只有一棵子树的情况下也要明确指出该子树是左子树还是右子树。

- 二叉树的节点的最大度为 2，而树中不限制节点的度数。

2. 二叉树的运算

二叉树的基本运算是遍历，其他运算可建立在遍历运算的基础上。

3. 二叉树的性质

二叉树具有以下性质。

(1) 二叉树第 i 层上的节点数目最多为 $2^{i-1}(i\geqslant1)$个。

(2) 深度为 k 的二叉树至多有 $2^k-1(k\geqslant1)$个节点。

(3) 在任意一棵二叉树中，若终端节点数为 n_0，度为 2 的节点数为 n_2，则 $n_0=n_2+1$。

(4) 具有 n 个节点的完全二叉树的深度为$[\log_2 n]+1$。

(5) 对一棵有 n 个节点的完全二叉树的节点按层次自左至右进行编号，则对任意节点 i 有以下性质。

- 若 $i=1$，则节点 i 是二叉树的根，无双亲；若 $i>1$，则其双亲为$\left[\dfrac{i}{2}\right]$。
- 若 $2i>n$，则节点 i 无左孩子；否则其左孩子为 $2i$。
- 若 $2i+1>n$，则节点 i 无右孩子；否则其右孩子为 $2i+1$。

若深度为 k 的二叉树有 2^k-1 个节点，则称其为满二叉树。

深度为 k、有 n 个节点的二叉树，当且仅当其每一个节点都与深度为 k 的满二叉树编号从 1 至 n 的节点一一对应时，称之为完全二叉树。

4. 二叉树的存储结构

1) 顺序存储结构

用一组地址连续的存储单元存储二叉树中的数据元素，必须把节点排成一个适当的线性序列，并且节点在这个序列中的相互位置能反映出节点之间的逻辑关系。

顺序存储结构用于完全二叉树时既简单又节省空间，而对于一般二叉树则不适用。因为在顺序存储结构中，以节点在存储单元中的位置来表示节点之间的关系，那么对于一般的二叉树来说，也必须按照完全二叉树的形式存储，也就是要添上一些实际并不存在的“虚节点”，这将造成空间的浪费。

2) 链式存储结构

由于二叉树中的节点包含有数据元素、左子树根、右子树根及双亲等信息，因此可以用三叉链表或二叉链表来存储二叉树，链表的头指针指向二叉树的根节点。

5. 二叉树的遍历

遍历是指按某种策略访问树中的每个节点，且仅访问一次。由于二叉树所具有的递归性质，一棵非空的二叉树可以看作由根节点、左子树和右子树三部分构成，因此若能依次遍历这三部分中的每个节点信息，也就遍历了整棵二叉树。按照遍历左子树要在遍历右子树之前进行的约定，根据访问根节点位置的不同，可得到二叉树的前序、中序和后序 3 种遍历方法。

遍历二叉树的基本操作就是访问节点，不论按照哪种次序遍历，对含有 n 个节点的二叉树，遍历算法的时间复杂度都为 $O(n)$。在最坏情况下，二叉树是有 n 个节点且深度为 n

的单枝树，遍历算法的空间复杂度也为 $O(n)$。

遍历二叉树的过程实质上是按一定规则，将树中的节点排成一个线性序列的过程，因此遍历操作得到的是树中节点的一个线性序列。在每一种序列中，有且仅有一个起始点和一个终节点，其余节点有且仅有唯一的直接前驱和直接后继。

对二叉树还可以进行层序遍历。层序遍历就是从树的根节点出发，首先访问第 1 层的树根节点，然后从左到右依次访问第 2 层上的节点，以此类推，自上而下、自左到右逐层访问树中各层上节点的过程。

6. 线索二叉树

若 n 个节点的二叉树采用链表作存储结构，则链表中含有 $n+1$ 个空指针域，利用这些空指针域来存放指向节点的前驱和后继信息。线索链表的节点结构如图 3-6 所示。

ltag	lchild	data	rchild	rtag

图 3-6　线索链表的节点结构

若二叉树的二叉链表采用图 8-6 所示的节点结构，则相应的链表称为线索链表，其中指向节点前驱、后继的指针称为线索，加上线索的二叉树称为线索二叉树。对二叉树以某种次序遍历使其变为线索二叉树的过程称为线索化。

7. 二叉树的应用：最优二叉树

霍夫曼树又称最优二叉树，是一类带权路径长度最短的树。

路径：是指从树中一个节点到另一个节点之间的通路，路径上的分支数目称为路径长度。

树的路径长度：是从树根到每一个叶子的路径长度之和。节点的带权路径长度为从该节点到树根之间的路径长度与该节点权的乘积。

树的带权路径长度：指树中所有叶子节点的带权路径长度之和，记为

$$\mathrm{WPL}=\sum_{i=1}^{n} w_i l_i$$

式中，n 为带权叶子节点的数目；w_i 为叶子节点的权值；l_i 为叶子节点到根的路径长度。

霍夫曼树是指权值为 $w_1, w_2, \cdots, w_n$ 的 n 个叶子节点的二叉树中带权路径长度最小的二叉树。

构造最优二叉树的霍夫曼算法如下。

(1) 根据给定的 n 个权值 $w_1, w_2, \cdots, w_n$ 构成 n 棵二叉树的集合 $F=\{T_1, T_2, \cdots, T_n\}$，其中每棵二叉树 T_i 中只有一个带权为 w_i 的根节点，其左右子树均空。

(2) 在 F 中选取两棵根节点的权值最小的树作为左右子树，构造一棵新的二叉树，置新构造二叉树的根节点的权值为其左、右子树根节点的权值之和。

(3) 从 F 中删除这两棵树，同时将新得到的二叉树加入到 F 中。

重复(2)、(3)，直到 F 中只含一棵树时为止。这棵树便是霍夫曼树。

8. 树和森林

1) 树的存储结构

- 树的双亲表示法：用一组地址连续的单元存储树的节点，并在每个节点中附设一个指示器，指示其双亲节点在该存储结构中的位置。显然这种表示对于求指定节点

的双亲或祖先都十分方便，但对于求指定节点的孩子及后代则需要遍历整个数组。

- 树的孩子表示法：在存储结构中用指针指示出节点的每个孩子，由于树中每个节点的子树数目不尽相同，因此在采用链式存储结构时可以考虑多重链表。
- 树的孩子兄弟表示法：又称二叉链表表示法。在链表的节点中设置两个指针域分别指向该节点的第一个孩子和下一个兄弟。利用这种存储结构便于实现树的各种操作。

2)　树和森林的遍历

(1)　树的遍历。树的遍历分为先根遍历和后根遍历两种。

- 先根遍历：先访问树的根节点，然后依次先根遍历根的各棵子树。对树的先根遍历等同于对转换所得的二叉树进行先序遍历。
- 后根遍历：先依次后根遍历树根的各棵子树，然后访问树根节点。树的后根遍历等同于对转换所得的二叉树进行中序遍历。

(2)　森林的遍历。森林的遍历分为前序遍历和后序遍历两种。

- 前序遍历森林：若森林非空，访问森林中第一棵树的根节点，前序遍历第一棵子树根节点的子树森林，再前序遍历除第一棵树之外剩余的树所构成的森林。
- 后序遍历森林：若森林非空，后序遍历森林中第一棵树的子树森林，访问第一棵树的根节点，后序遍历除第一棵树之外剩余的树所构成的森林。

3)　树、森林与二叉树的转换

(1)　树、森林转换为二叉树。利用树的孩子兄弟表示法可导出树与二叉树的对应关系，在树的孩子兄弟表示法中，从物理结构上看与二叉树的二叉链表表示法相同，因此就可以用这种同一存储结构的不同解释将一棵树转换为一棵二叉树。

将一个森林转换为一棵二叉树的方法是：先将森林中的每一棵树转换为二叉树，再将第一棵树的根作为转换后的二叉树的根，第一棵树的左子树作为转换后二叉树根的左子树，第二棵树作为转换后二叉树根的右子树，第三棵树作为转换后二叉树根的右子树的右子树，以此类推，森林就可以转换为一棵二叉树。

(2)　二叉树转换为树和森林。若二叉树非空，则二叉树根及其左子树为第一棵树的二叉树形式，二叉树根的右子树又可以看作一个由森林转换后的二叉树，应用同样的方法，直到最后产生一棵没有右子树的二叉树为止，这样就得到了一个森林。为了进一步得到树，可用树的二叉链表表示的逆方法，即节点的右子树的根、右子树的右子树的根……找出原本是同一个双亲的兄弟。二叉树转换为树或森林是唯一的。

3.3.2　典型例题分析

例 1　一个高度为 h 的满二叉树的节点总数为 2^h-1，从根节点开始，自上而下、同层次节点从左至右，对节点按照顺序依次编号，即根节点编号为 1，其左、右孩子节点编号分为 2 和 3，再下一层从左到右的编号为 4、5、6、7，依次类推。那么，在一棵满二叉树中，对于编号为 m 和 n 的两个节点，若 $n=2m+1$，则 (64) 节点。(2013 年上半年试题 64)

(64) A. m 是 n 的左孩子　　B. m 是 n 的右孩子

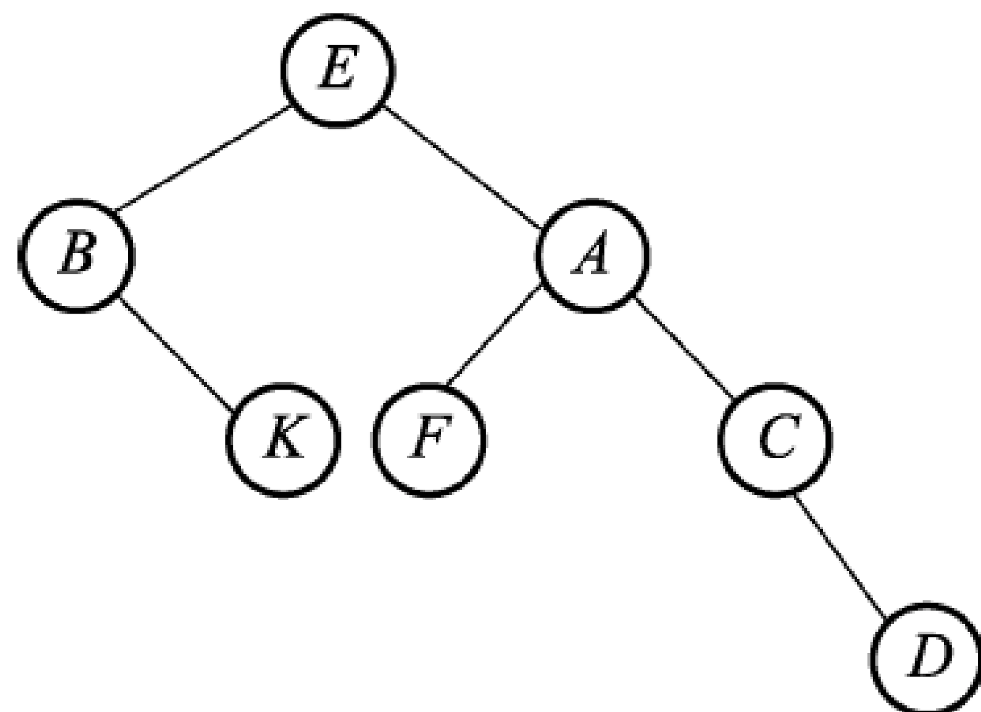
C. n 是 m 的左孩子　　D. n 是 m 的右孩子

解析： 由于该二叉树为满二叉树，且根节点编号从 1 开始，由满二叉树的性质可知父节点 m 和右孩子之间的关系为 $n=2m+1$。

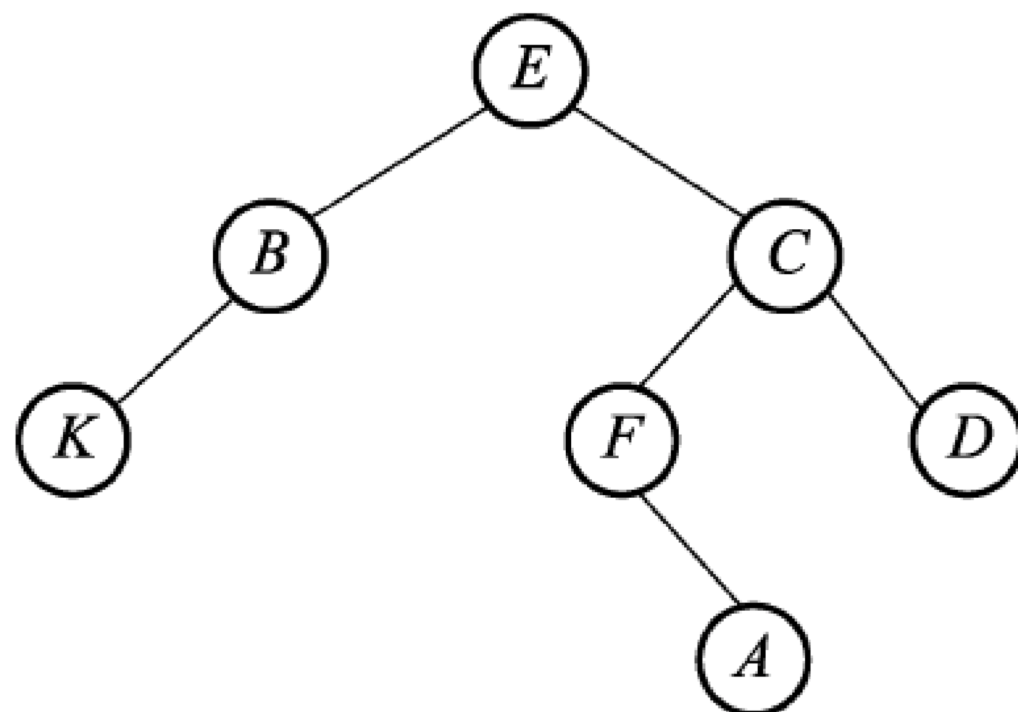
答案： D

例 2 若某二叉树的后序遍历序列为 *KBFDCAE*，中序遍历序列为 *BKFEACD*，则该二叉树为 (58) 。(2012 年下半年试题 58)

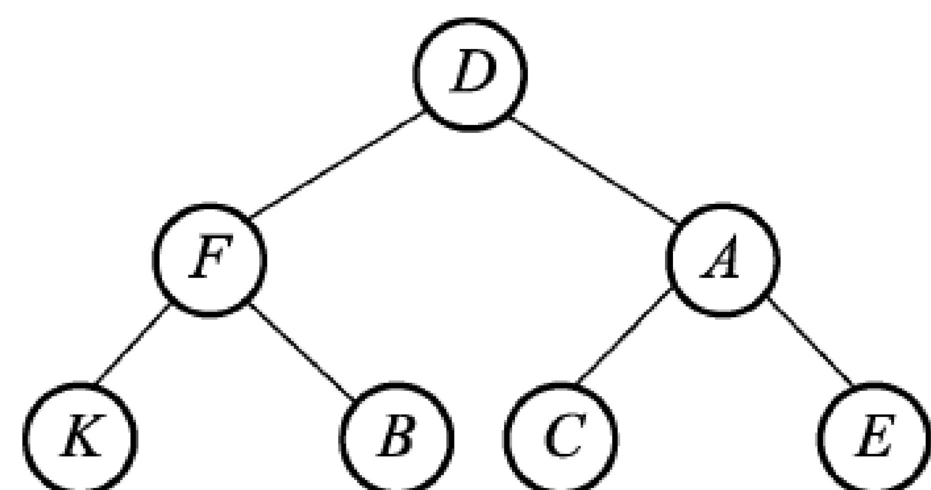
(58) A.



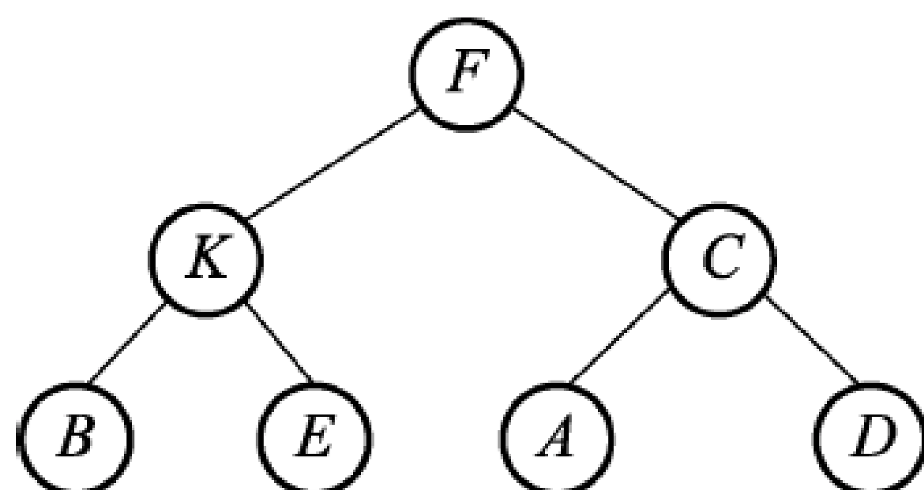
B.



C.



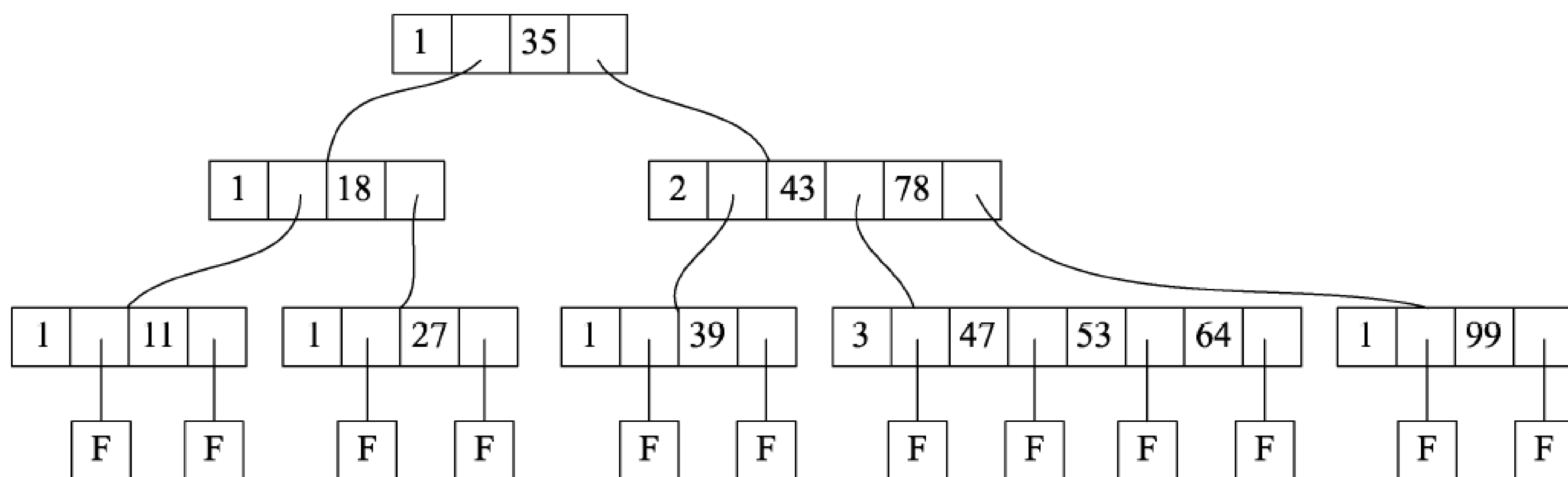
D.



解析： 本题考查二叉树的遍历算法，根据中序遍历序列和另一种遍历序列的结果，可以确定该二叉树。后序遍历是按照左子树、右子树、根节点的顺序进行遍历，中序遍历是按照左子树、根节点、右子树的顺序进行遍历。*E* 为根节点，*K* 为 *B* 的右子树，因此应选 A 项描述的二叉树。

答案： A

例 3 下图所示为一棵 *N* 阶 *B*-树，*N* 最有可能的值为 (61) 。(2012 年下半年试题 61)



(61) A．1　　B．2　　C．3　　D．4

解析： 一棵 *N* 阶 B⁻树为满足以下特性的 *N* 叉树。

① 树中每个节点至多有 *N* 棵子树。

② 若根节点不是叶子节点，则至少有两棵子树。

③ 除根之外的所有非终端节点至少有$\left\lceil \frac{N}{2} \right\rceil$棵子树;

④ 所有的非终端节点中包含下列数据信息($n, A_0, K_1, A_1, K_2, A_2, \cdots, K_n, A_n$)。其中，$K_i$($i$=1, 2, …, n)为关键字(如 3、47、53、63)，且$K_i < K_{i+1}$，A_i(i=0、1、2、…、n)为指向子树根节点的指针，n 为节点中关键字的个数，且$\left\lceil \frac{N}{2} \right\rceil - 1 \leqslant n \leqslant N - 1$。

⑤ 所有的叶子节点都出现在同一层次上，并且不带信息。

由上图可知，N 最有可能的值为 4。

答案：D

例 4　若 n_2、n_1、n_0 分别表示一个二叉树中度为 2、度为 1 和叶子节点的数目(节点的度定义为节点的子树数目)，则对于任何一个非空的二叉树，__(59)__。(2012 年上半年试题 59)

(59) A. n_2一定大于 n_1　　　B. n_1一定大于 n_0

C. n_2一定大于 n_0　　　D. n_0一定大于 n_2

解析：由二叉树的性质可知，度为 0 的节点比度为 2 的节点数多 1，即 n_0=n_2+1，因此 n_0一定大于 n_2。

答案：D

例 5　若一棵二叉树的高度(即层数)为 h，则该二叉树__(59)__。(2016 年上半年试题 59)

(59) A．有 2h 个节点　　　B．有 2h−1 个节点

C．最少有 2h−1 个节点　　　D．最多有 2h−1 个节点

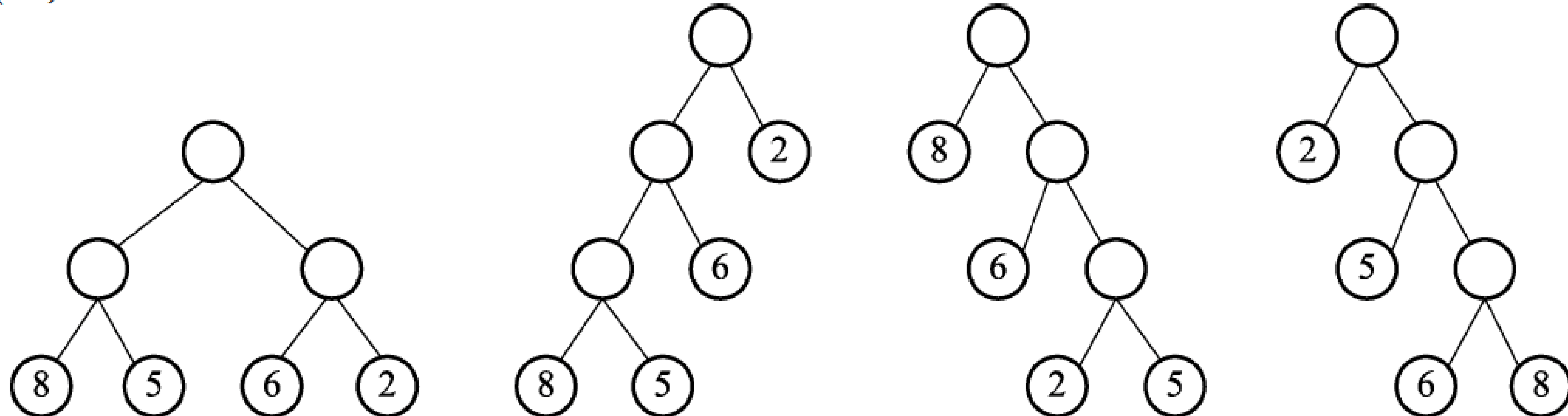
解析：一棵高度为 h 的二叉树，节点数最多时，即为满二叉树。

而高度为 h 的满二叉树有 2h−1 个节点，所以一棵二叉树的高度(即层数)为 h，则它最多有 2h−1 个节点。

答案：D

例 6　__(61)__是由权值集合{8,5,6,2}构造的哈夫曼树(最优二叉树)。(2011 年下半年试题 61)

(61) A.　　　B.　　　C.　　　D.



解析：哈夫曼树是带权路径最短的树。选项 A、B、C、D 这 4 棵树的带权路径长度分别如下。

选项 A：8*2+5*2+6*2+2*2=42

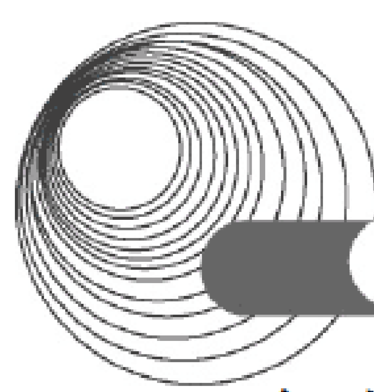
选项 B：8*3+5*3+6*2+2=53

选项 C：8+6*2+2*3+5*3=41

选项 D：2+5*2+6*3+8*3=54

答案：C

例 7　在__(59)__中，任意一个节点的左、右子树的高度之差的绝对值不超过 1。(2011



年上半年试题 59)

(59) A．完全二叉树　B．二叉排序树　C．线索二叉树　D．最优二叉树

解析：对于完全二叉树，若设二叉树的高度为 h，除第 h 层外，其他各层($1\sim h-1$)的节点数都达到最大个数，第 h 层所有的节点都连续集中在最左边，这就是完全二叉树。在完全二叉树中，任意一个节点的左、右子树的高度之差的绝对值不超过 1。

二叉排序树(Binary Sort Tree)又称二叉查找树。它或者是一棵空树；或者是具有下列性质的二叉树：①若左子树不空，则左子树上所有节点的值均小于它的根节点的值；②若右子树不空，则右子树上所有节点的值均大于它的根节点的值；③左、右子树也分别为二叉排序树。对于二叉排序树，由于左子树或右子树可能为空，不能保证每一个节点的左、右子树的高度之差的绝对值不超过 1。

按照某种遍历方式对二叉树进行遍历，可以把二叉树中所有节点排序为一个线性序列。在该序列中，除第一个节点外每个节点有且仅有一个直接前驱节点；除最后一个节点外每一个节点有且仅有一个直接后继节点。这些指向直接前驱节点和指向直接后续节点的指针被称为线索(Thread)，加了线索的二叉树称为线索二叉树。线索二叉树只是加了线索的二叉树，对节点的排列没有要求，不能保证每一个节点的左、右子树的高度之差的绝对值不超过 1。

给定 n 个权值作为 n 个叶子节点，构造一棵二叉树，若带权路径长度达到最小，称这样的二叉树为最优二叉树，也称为哈夫曼树，不能保证每一个节点的左、右子树的高度之差的绝对值不超过 1。

答案：A

例 8　已知一个文件中出现的各个字符及其对应的频率如下表所示。若采用定长编码，则该文件中字符的码长应为 (64) 。若采用 Huffman 编码，则字符序列”face”的编码应为(65)。(2014 年下半年试题 64、65)

字符	a	b	c	d	e	f
频率(%)	45	13	12	16	9	5

(64) A．2　　B．3　　C．4　　D．5

(65) A．110001001101　　B．001110110011

C．101000010100　　D．010111101011

解析：(1)　有 6 个不同字母，需要采用 3 位二进制进行编码。

(2)　哈夫曼静态编码。它对需要编码的数据进行两遍扫描：第一遍统计原数据中各字符出现的频率，利用得到的频率值创建哈夫曼树，并必须把树的信息保存起来，即把字符 0～255(2^8=256)的频率值以 2～4B 的长度顺序存储起来(用 4B 的长度存储频率值，频率值的表示范围为 $0\sim2^{32}-1$，这已足够表示大文件中字符出现的频率了)，以便解压时创建同样的哈夫曼树进行解压；第二遍则根据第一遍扫描得到的哈夫曼树进行编码，并把编码后得到的码字存储起来。

答案：(64) B　　(65) A

例 9　已知一棵度为 3 的树(一个节点的度是指其子树的数目，树的度是指该树中所有节点的度的最大值)中有 5 个度为 1 的节点，4 个度为 2 的节点，2 个度为 3 的节点，那么，该树中的叶子节点数目为 (61) 。(2010 年下半年试题 61)

(61) A．10　　　B．9　　　C．8　　　D．7

解析：树的节点总数为 5+4×2+2×3+1=20，叶子节点数为 20−5−4−2=9。

答案：B

例 10　若用 n 个权值构造一棵最优二叉树(哈夫曼树)，则该二叉树的节点总数为 (59) 。(2010 年上半年试题 59)

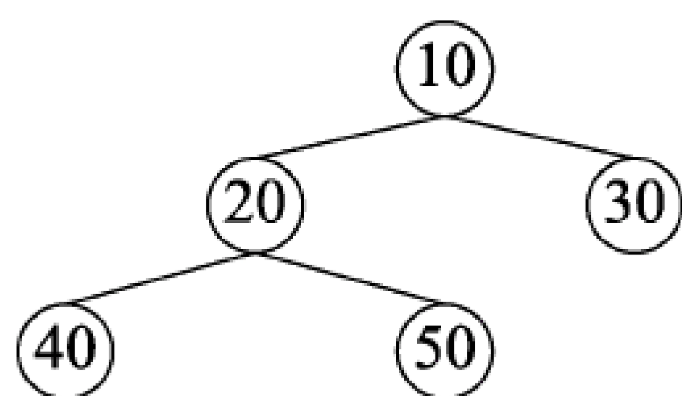
(59) A．$2n$　　　B．$2n-1$　　　C．$2n+1$　　　D．$2n+2$

解析：二叉树具有以下性质：度为 2 的节点(双分支节点)数比度为 0 的节点(叶子节点)数正好少 1。而根据最优二叉树(哈夫曼树)的构造过程可知，最优二叉树中只有度为 2 和 0 的节点，因此，其节点总数为 $2n-1$。

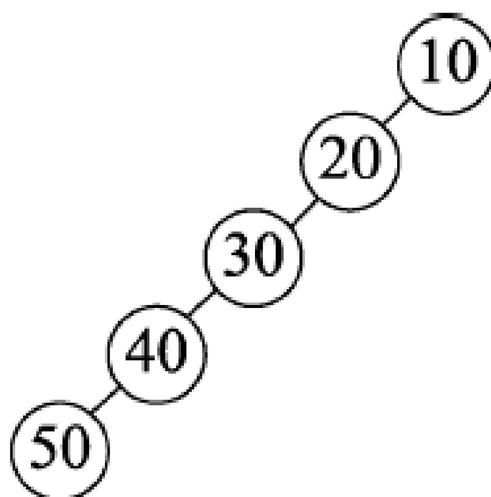
答案：B

例 11　用关键字序列 10、20、30、40、50 构造的二叉树排序(二叉查找树)为 (63) 。(2010 年上半年试题 63)

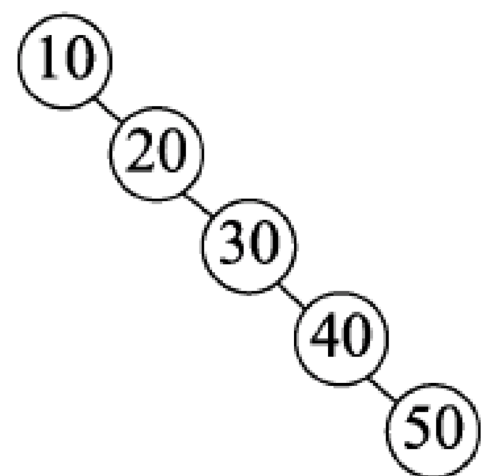
(63) A.



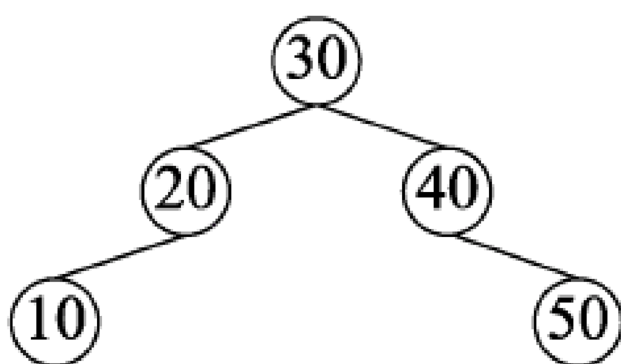
B.



C.



D.



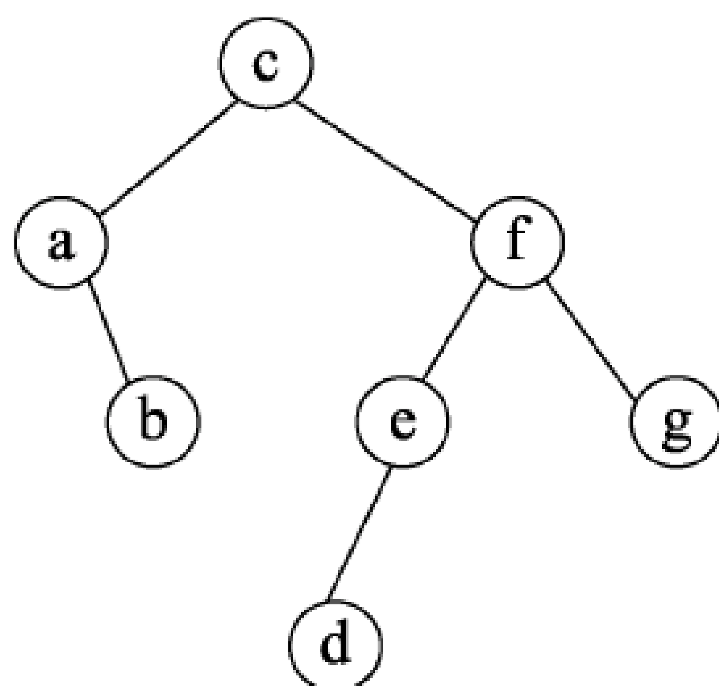
解析：根据关键字序列构造二叉排序树的基本过程是，若需插入的关键字大于树根，则插入到右子树上，若小于树根，则插入到左子树上，若为空，则作为数根节点。

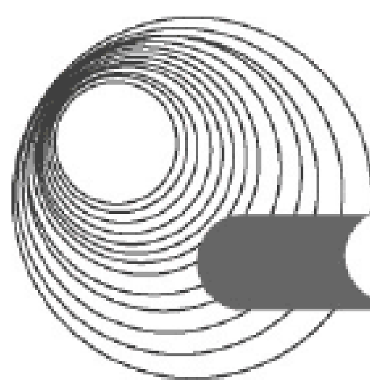
答案：C

例 12　某二叉树的先序遍历序列为 c、a、b、f、e、d、g，中序遍历序列为 a、b、c、d、e、f、g，则该二叉树是 (59) 。(2015 年上半年试题 59)

(59) A. 完全二叉树　B. 最优二叉树　C. 平衡二叉树　D. 满二叉树

解析：由先序遍历序列可知，c 为二叉树的根节点；结合中序遍历序列可知，a、b 组成左子树，d、e、f、g 组成右子树。继而判断 a 为左子树的根节点，b 为其右孩子节点；f 为右子树的根节点，d、e 为其左子树，g 为其右孩子节点；e 为 f 的左子树的根节点，d 为 e 的左孩子节点。因此，可得以下二叉树：





平衡二叉树是具有这样性质的二叉树：左子树和右子树都是平衡二叉树，且左子树和右子树的高度之差的绝地址不超过 1。显然，上述的二叉树满足这样的性质。

答案：C

例 13 下面关于二叉树的叙述，正确的是 (61) 。(2009 年上半年试题 61)

(61) A．完全二叉树的高度 h 与其节点数 n 之间存在确定的关系

B．在二叉树的顺序存储和链式存储结构中，完全二叉树更适合采用链式存储结构

C．完全二叉树中一定不存在度为 1 的节点

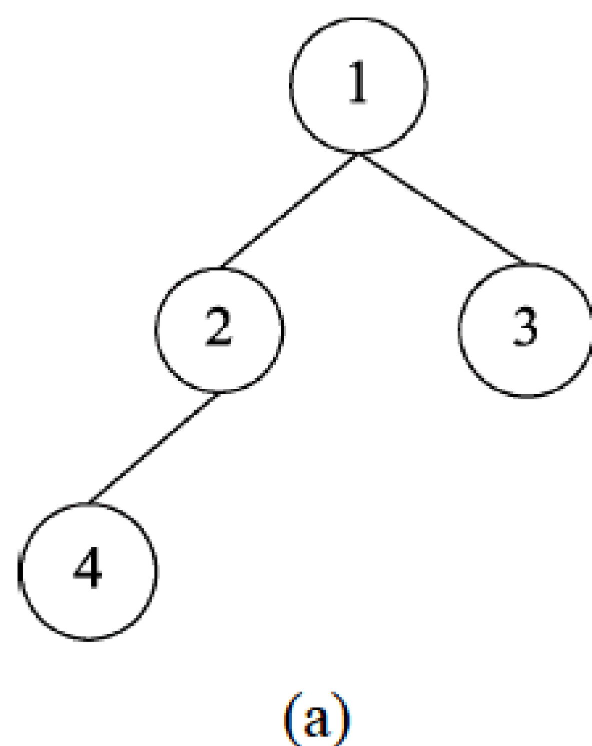
D．完全二叉树中必定有偶数个叶子节点

解析：如果一棵具有 n 个节点的深度为 k 的二叉树，它的每一个节点都与深度为 k 的满二叉树中编号为 1～n 的节点一一对应，称之为完全二叉树。由其性质：具有 n 个节点的完全二叉树的深度为$\lfloor \log_2 n \rfloor+1$。可知 A 正确。

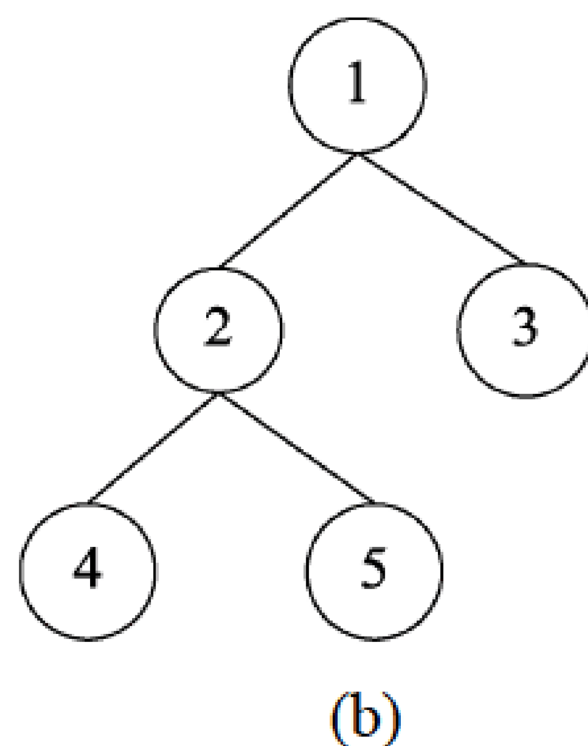
对于 B，按照顺序存储结构的定义，用一组地址连续的存储单元依次自上而下、自左至右存储完全二叉树的节点元素。因此用顺序存储结构更利于完全二叉树的节点访问。

对于 C，图(a)所示为完全二叉树，但它有度为 1 的节点，即 2 号节点。

对于 D，图(b)所示为完全二叉树，但它有奇数个叶子节点。



(a)



(b)

答案：A

3.3.3 同步练习

1. 表达式$(a-b)*(c+5)$的后缀式是______。

A．$a\ b\ c\ 5 +*-$　　B．$a\ b-c+5\ *$　　C．$a\ b\ c-*\ 5+$　　D．$a\ b-c\ 5+*$

2. 一个具有 m 个节点的二叉树，其二叉链表节点(左、右孩子指针分别用 left 和 right 表示)中的空指针总数必定为 (1) 个。为形成中序(先序、后序)线索二叉树，现对该二叉链表所有节点进行以下操作：若节点 p 的左孩子指针为空，则将该左指针改为指向 p 在中序(先序、后序)遍历序列的前驱节点；若 p 的右孩子指针为空，则将该右指针改为指向 p 在中序(先序、后序)遍历序列的后继节点。假设指针 s 指向中序(先序、后序)线索二叉树中的某节点，则 (2) 。

(1) A．$m+2$　　B．$m+1$　　C．m　　D．$m-1$

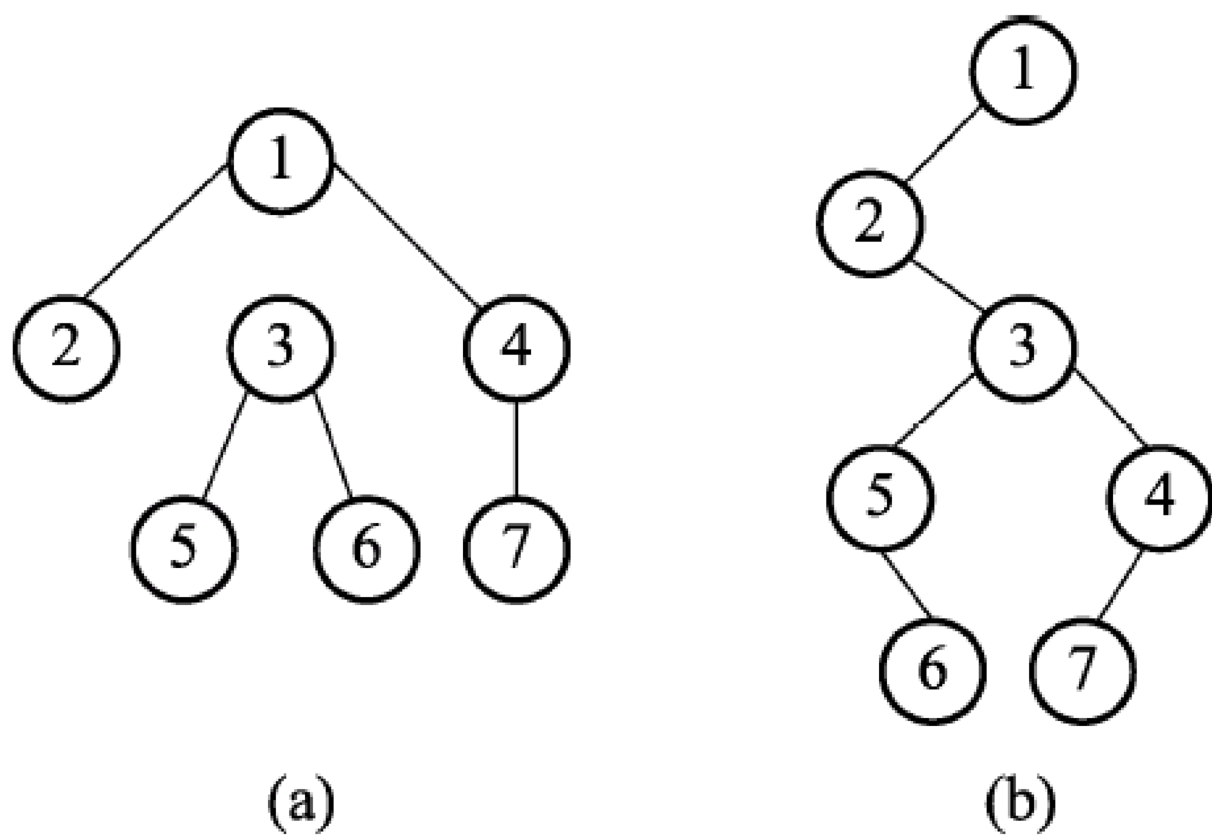
(2) A．$s\rightarrow$ right 指向的节点一定是 s 所指节点的直接后继节点

B．$s\rightarrow$ left 指向的节点一定是 s 所指节点的直接前驱节点

C．从 s 所指节点出发的 right 链可能构成环

D．s 所指节点的 left 和 right 指针一定指向不同的节点

3. 若将某有序树 T 转换为二叉树 T_1，则 T 中节点的后(根)序列就是 T_1 中节点的_____遍历序列。例如，图(a)所示的有序树转化为二叉树后如图(b)所示。



A．先序　　B．中序　　C．后序　　D．层序

4. 表达式“X=A+B×(C−D)/E”的后缀表示形式可以为______(运算符优先级相同时，遵循左结合的原则)。

A．XAB + CDE/−×=　　B．XA+BC−DE/×=

C．$XABCD$−×E/+=　　D．$XABCDE$+×−/=

5. 对于二叉查找树(Binary Search Tree)，若其左子树非空，则左子树上所有节点的值均小于根节点的值；若其右子树非空，则右子树上所有节点的值均大于根节点的值；左、右子树本身就是两棵二叉查找树。因此，对任意一棵二叉查找树进行 (1) 遍历可以得到一个节点元素的递增序列。在具有 n 个节点的二叉查找树上进行查找运算，最坏情况下的算法复杂度为 (2) 。

(1) A．先序　　B．中序　　C．后序　　D．层序

(2) A．$O(n^2)$　　B．$O(n\log_2 n)$　　C．$O(\log_2 n)$　　D．$O(n)$

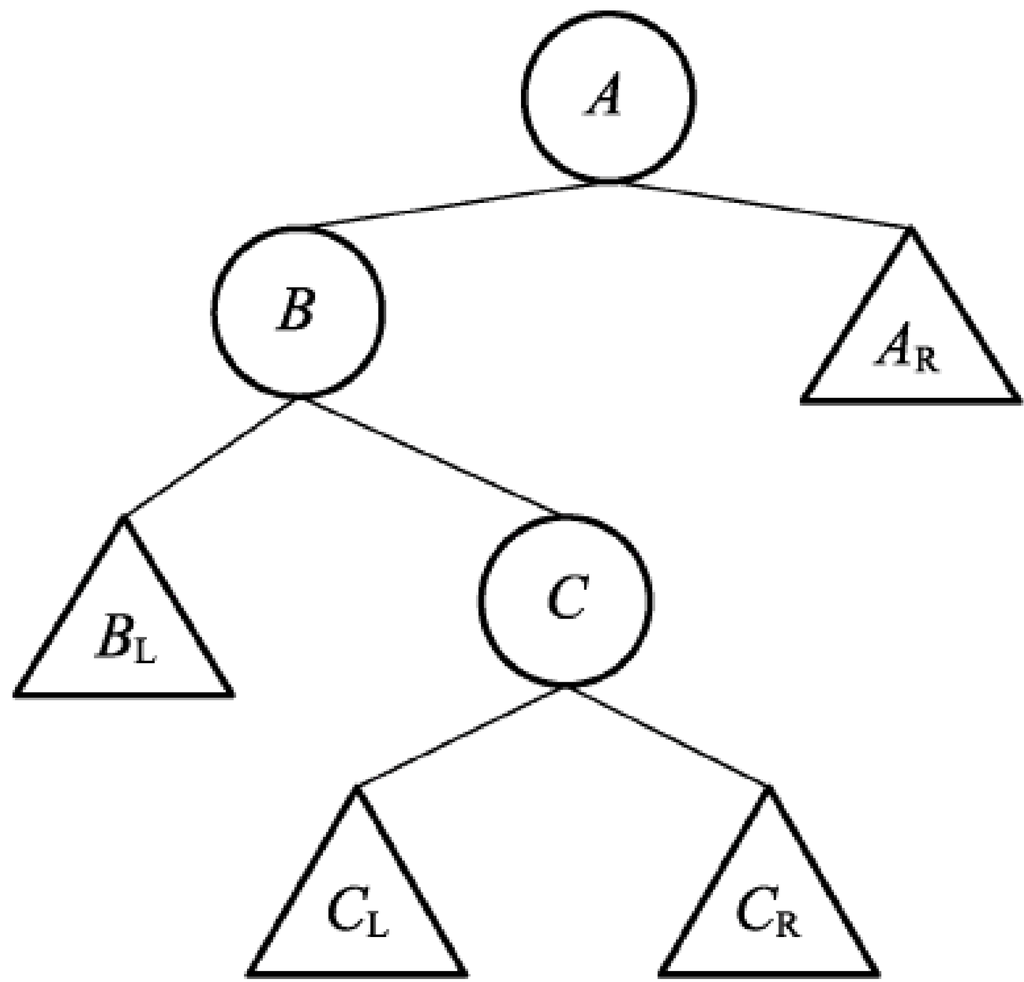
6. 表达式“(a+b)* (c−d)”的后缀表示为_____。

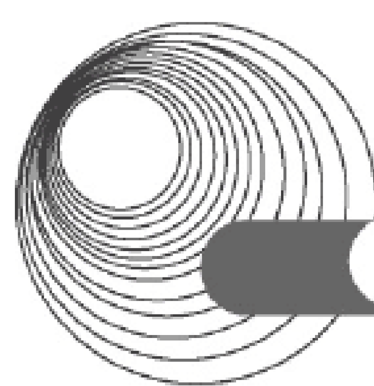
A．ab+cd−*　　B．$abcd$+−*　　C．ab+*cd−　　D．$abcd$*+−

7. 已知某二叉树的中序列为 $CBDAEFI$、先序列为 $ABCDEFI$，则该二叉树的高度为_____。

A．2　　B．3　　C．4　　D．5

8. 下图所示平衡二叉树(树中任一节点的左、右子树高度之差不超过 1)中，节点 A 的右子树 A_R 高度为 h，节点 B 的左子树 B_L 高度为 h，节点 C 的左子树 C_L、右子树 C_R 高度都为 h−1。若在 C_R 中插入一个节点并使得 C_R 的高度增加 1，则该二叉树_____。





A. 以 B 为根的子二叉树变为不平衡

B. 以 C 为根的子二叉树变为不平衡

C. 以 A 为根的子二叉树变为不平衡

D. 仍然是平衡二叉树

3.3.4 同步练习参考答案

1. D
2. (1) B　　(2) C
3. B
4. C
5. (1) B　　(2) D
6. A
7. C
8. C

3.4 图

3.4.1 考点辅导

3.4.1.1 图的定义

图 G 是由两个集合 V 和 E 构成的二元组，记作 $G=(V,E)$，其中 V 是图中顶点的非空有限集合，E 是图中边的有限集合。从数据结构的逻辑关系来看，图中任一顶点都有可能与图中其他顶点有关系，而图中所有顶点都有可能与某一顶点有关系。在图中，数据结构中的数据元素用顶点表示，数据元素之间的关系用边表示。

(1) 有向图。若图中每条边都是有方向的，则称 G 为有向图。顶点间的关系用 $<v_i, v_j>$ 表示，它说明从 v_i 到 v_j 的一条有向边(也称为弧)，v_i 是有向边的起点，称为弧尾；v_j 是有向边的终点，称为弧头。

(2) 无向图。若图中的每条边都是无方向的，则顶点 v_i 和 v_j 之间的边用 (v_i, v_j) 表示。

(3) 无向完全图。若一个无向图具有 n 个顶点，而每一个顶点与其他 $n-1$ 个顶点之间都有边，则称之为无向完全图。显然，含有 n 个顶点的无向完全图共有 $n(n-1)/2$ 条边。

(4) 有向完全图。有 n 个顶点的有向完全图中弧的数目为 $n(n-1)$，即任何两个不同顶点之间都有方向相反的两条弧存在。

(5) 度、入度和出度。顶点的度是指关联于该顶点的边的数目，记为 $D(v)$。若 G 为有向图，顶点的度表示该顶点的入度和出度之和。顶点的入度是指以该顶点为终点的有向边的数目，而顶点的出度是指以该顶点为起点的有向边的数目，分别记为 $\mathrm{ID}(v)$和 $\mathrm{OD}(v)$。无

论是有向图还是无向图，顶点数 n、边数 e 与各顶点的度之间有

$$e = \frac{1}{2}\sum_{i=1}^{n} D(v_i)$$

(6) 路径。在无向图 G 中，从顶点 v_p 到顶点 v_q 的路径是指存在一个顶点序列 v_p，$v_{i1}, v_{i2}, \cdots, v_{in}, v_q$，使得 (v_{i1}, v_p)，(v_{i1}, v_{i2})，…，(v_{in}, v_q) 均属于 $E(G)$。

(7) 子图。对于两个图 $G=(V,E)$ 和 $G'=(V',E')$，如果 V' 是 V 的子集，E' 是 E 的子集，则称 G' 为 G 的子图。

(8) 连通图。在无向图 G 中，若从顶点 v_i 到顶点 v_j 有路径，则称顶点 v_i 和顶点 v_j 是连通的。如果无向图 G 中任意两个顶点都是连通的，则称其为连通图。无向图 G 的极大连通子图称为 G 的连通分量。

(9) 强连通图。在有向图 G 中，如果对于每一对顶点 v_i，v_j 且 $v_i \neq v_j$，从顶点 v_i 到顶点 v_j 和从顶点 v_j 到顶点 v_i 都存在路径，则称图 G 为强连通图。也就是说，如果 $V(G)$ 中任意两个不同的顶点 v_i 和 v_j，都存在从 v_i 到 v_j 以及从 v_j 到 v_i 的路径，则称 G 是强连通图。

(10) 网。边(或弧)带权值的图称为网。

(11) 生成树。一个连通图的生成树是一个极小的连通子图，它包含图中的全部顶点，但只有构成一棵树的 $n-1$ 条边。

(12) 有向树和生成森林。如果一个有向图恰有一个顶点的入度为 0，其余顶点的入度均为 1，则是一棵有向树。有向图的生成森林由若干棵有向树组成，含有图中全部顶点，但只有足以构成若干棵不相交的有向树的弧。

3.4.1.2　存储结构

1. 邻接矩阵表示法

对于具有 n 个顶点的图 $G(V, E)$ 来说，其邻接矩阵是一个 n 阶方阵，且满足

$$A[i][j] = \begin{cases} 1, 若(v_i, v_j)或<v_i, v_j>是E中的边 \\ 0, 若(v_i, v_j)或<v_i, v_j>不是E中的边 \end{cases}$$

由邻接矩阵的定义可知，无向图的邻接矩阵是对称的，有向图的邻接矩阵就不一定对称了。借助邻接矩阵易判定任意两个顶点之间是否有边(或弧)相连，并且容易求得各个顶点的度。

网(赋权图)的邻接矩阵可定义为

$$A[i][j] = \begin{cases} W_{ij}, 若(v_i, v_j)或<v_i, v_j>是E中的边 \\ \infty, 若(v_i, v_j)或<v_i, v_j>不是E中的边 \end{cases}$$

2. 邻接链表表示法

邻接链表指的是为图的每个顶点建立一个单链表，第 i 个单链表中的节点表示依附于顶点 v_i 的边(对于有向图是以 v_i 为尾的弧)。邻接链表中的节点有表节点和表头节点两种类型。

邻接矩阵和邻接链表表示法对有向图和无向图都适用。

3.4.1.3 图的遍历

1. 深度优先遍历

从图 G 中任一个顶点 v 出发，深度优先遍历(DFS)的算法步骤如下。

(1) 设立搜索指针 p，使 p 指向顶点 v。

(2) 访问 p 顶点，并使 p 指向与 p 顶点相邻接的且尚未被访问过的顶点。

(3) 若 p 不空，则重复步骤(2)；否则执行步骤(4)。

(4) 沿着刚才访问的次序、方向回溯到一个尚有邻接顶点且未被访问过的顶点，并使 p 指向这个未被访问的邻接顶点，然后重复步骤(2)，直至所有的顶点均被访问为止。

这个算法的特点是尽可能先对纵深方向搜索，因此可以很容易得到其遍历的递归算法。

深度优先遍历图的过程实质上是对某个顶点查找其邻接节点的过程，其耗费的时间取决于所采用的存储结构。当图用邻接矩阵表示时，查找所有顶点的邻接点所需时间为 $O(n^2)$。若以邻接表作为图的存储结构，则需要 $O(e)$的时间复杂度查找所有顶点的邻接点。因此，当以邻接表作为存储结构时，深度优先搜索遍历图的时间复杂度为 $O(n+e)$。

2. 广度优先遍历(BFS)

广度优先遍历(BFS)的遍历过程是：假设从图中某一个顶点 v 出发，在访问 v 之后依次访问 v 的各个未被访问过的邻接点，然后分别从这些邻接点出发依次访问它们的邻接点，并使“先被访问的顶点的邻接点”先于“后被访问的顶点的邻接点”被访问，直至图中所有已被访问过的顶点的邻接点都被访问到。若此时还有未被访问的顶点，则另选其中一个作为起点，重复上述过程，直至图中所有的顶点都被访问到为止。

广度优先遍历图的特点是尽可能先进行横向搜索，即最先访问的顶点的邻接点亦先被访问。

3.4.1.4 生成树和最小生成树

1. 生成树

设图 $G=(V, E)$是个连通图，当从图中任一个顶点出发遍历图 G 时，将边集 $E(G)$分为两个集合，即 $A(G)$和 $B(G)$。其中 $A(G)$是遍历时所经过的边的集合，$B(G)$是遍历时未经过的边的集合。$G_1=(V, A)$是图 G 的子图，称子图 G_1 为连通图 G 的生成树。

图的生成树不是唯一的。选择不同的存储方式，从不同的顶点出发，可以得到不同的生成树。对于非连通图而言，每个连通分量中的顶点集和遍历时走过的边集一起构成若干棵生成树，把它们称为非连通图的生成树森林。

2. 最小生成树

对于连通网来说，边是带权值的，生成树的各边也带权值，如果把生成树各边的权值总和称为生成树的权，则把权值最小的生成树称为最小生成树。

构造生成树有多种算法，其中多数算法利用了最小生成树的 MST 性质：假设 $G=(V, E)$是一个连通图，U 是顶点集 V 的一个非空子集。若(u,v)是一条最小权值的边，其中 $u \in U, v \in V-U$，则必存在一棵包含边(u,v)的最小生成树。

① 普里姆算法。它的时间复杂度为$O(n^2)$，与图中的边数无关，因此该算法适合于求边稠密的网中的最小生成树。

② 克鲁斯卡尔算法。它的时间复杂度为$O(e\log_2 e)$，与图中的顶点数无关，因此该算法适合于求边稀疏的网的最小生成树。

3.4.1.5　拓扑排序和关键路径

1. AOV 网

在有向图中，若一顶点表示活动，用有向边表示活动之间的优先关系，则称这样的有向图为以顶点表示活动的网，简称 AOV 网。

在 AOV 网中不应出现有向环。不存在回路的 AOV 网称为有向无环图或 DAG 图。检测的方法是对有向图构造其从顶点开始的拓扑有序序列，若图中所有顶点都在它的拓扑有序序列中，则该 AOV 网中必定不存在环。

2. 拓扑排序及其算法

拓扑排序是将 AOV 网中所有顶点排成一个线性序列，该序列满足：若在 AOV 网中从顶点v_i到v_j有一条路径，则在该线性序列中，顶点v_i必然在顶点v_j之前。拓扑排序即指对 AOV 网构造拓扑序列的操作。

对 AOV 网进行拓扑排序的方法如下。

(1) 在 AOV 网中选择一个入度为零的顶点且输出它。

(2) 从网中删除该顶点及与该顶点有关的所有边。

(3) 重复上述两步，直至网中不存在入度为零的顶点为止。

若在 AOV 网中考察各顶点的出度，并按下列步骤进行排序，则称为逆拓扑排序。

(1) 在 AOV 网中选择一个没有后继的顶点且输出它。

(2) 从网中删除该顶点，并删去所有到达该顶点的弧。

(2) 重复上述两步，直至网中不存在出度为零的顶点为止。

拓扑排序的时间复杂度为$O(n+e)$。

3. AOE 网

若在带权有向图G中以顶点表示事件，以有向边表示活动，边上的权值表示该活动持续的时间，则这种带权有向图称为用边表示活动的网，简称 AOE 网。

AOE 网中不应存在有向回路。

4. 关键路径和关键活动

从源点到汇点的路径中，长度最长的路径称为关键路径。关键路径上的所有活动均是关键活动。如果任何一项关键活动没有按期完成，则会影响整个工程的进度，而提高关键活动的速度可以缩短整个工程的工期。假设在n个顶点的 AOE 网中，顶点v_0表示源点，顶点v_{n-1}表示汇点，则计算关键活动及关键路径时可引入以下术语。

- 顶点事件的最早发生时间$v_e(j)$。它是指从源点v_0到v_j的最长路径长度(时间)。
- 顶点事件的最晚发生时间$v_l(i)$。它是指在不推迟整个工程完成日期的前提下，事件v_i所允许的最晚发生时间。
- 活动的最早开始时间$e(k)$。表示活动a_k最早可开工时间。
- 活动的最晚开始时间$l(k)$。它是指在不推迟整个工程完成日期的前提下，允许该

活动最晚开始的时间。

3.4.1.6 最短路径

1. 单源点最短路径

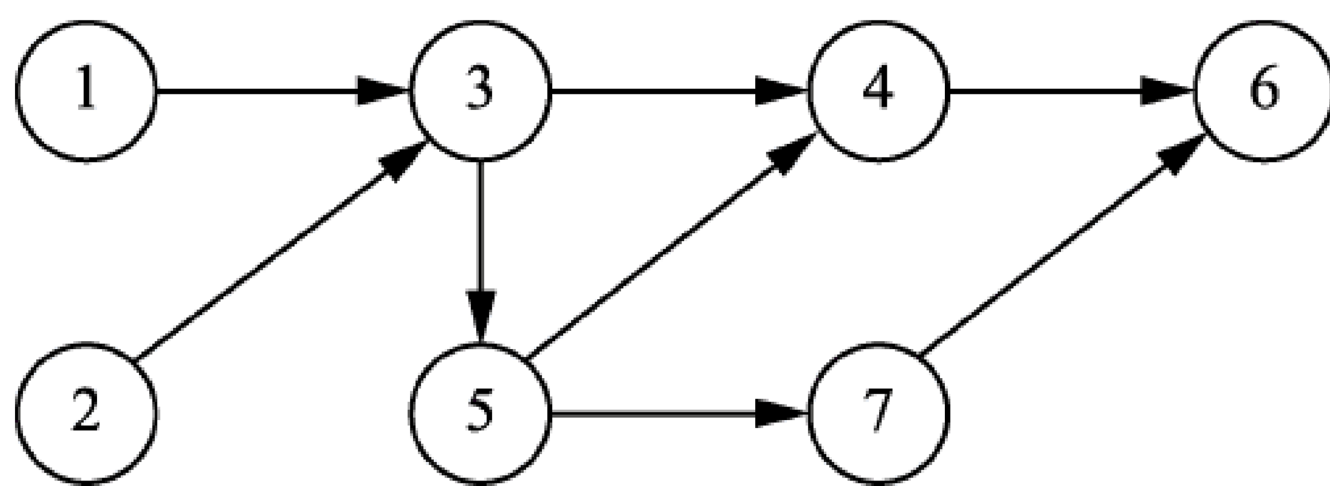
单源点最短路径是指给定带权有向图 G 和源点 v，求从 v 到 G 中其余各顶点的最短路径。迪杰斯特拉提出了按路径长度递增的次序产生最短路径的算法。

2. 每对顶点间的最短路径

若每次以一个顶点为源点，重复执行迪杰斯特拉算法 n 次，便可求得网中每一对顶点之间的最短路径。弗洛伊德提出了求最短路径的算法，该算法在形式上要简单一些。

3.4.2 典型例题分析

例 1 拓扑排序是将有向图中所有顶点排成一个线性序列的过程，并且该序列满足：若在 AOV 网中从顶点 V_i～V_j 有一条路径，则顶点 V_i 必然在顶点 V_j 之前。对于下图所示的有向图，<u>(60)</u>是其拓扑序列。(2012 年下半年试题 60)



(60) A．1234576　　B．1235467　　C．2135476　　D．2134567

解析：对 AOV 网进行拓扑排序的方法如下。

① 在 AOV 网中选择一个入度为 0(没有前驱)的顶点且输出它。

② 从网中删除该顶点及与该顶点有关的所有边。

③ 重复上述两步，直至网中不存在入度为零的顶点为止。

本题中只有序列“2135476”是其拓扑序列。

答案：C

例 2 从存储空间的利用率角度来看，以下关于数据结构中图存储的叙述，正确的是<u>(60)</u>。(2012 年上半年试题 60)

(60) A. 有向图适合采用邻接矩阵存储，无向图适合采用邻接表存储

B. 无向图适合采用邻接矩阵存储，有向图适合采用邻接表存储

C. 完全图适合采用邻接矩阵存储

D. 完全图适合采用邻接表存储

解析：邻接矩阵是用矩阵来指出顶点和顶点之间是否存在着关系。如果图有 n 个节点，则需要用 n^2 个元素来表示顶点间的关系。

邻接表是图的一种链式存储结构。在邻接表中，图中的每一个顶点都需要建立一个单链表，第 i 个单链表中的节点表示依附于顶点 v_i 的边。对于无向图，若无向图有 n 个顶点、e 条边，则它的邻接表需要 n 个头节点和 $2e$ 个表节点。对于有向图，若有 n 个顶点、e 条边，

则它的邻接表需要 n 个头节点和 e 个表节点。当 $e<<n(n-1)/2$ 时，采用邻接表表示图比用矩阵节省空间。可见，完全图适合采用邻接矩阵存储。

答案： C

例 3　无向图中一个顶点的度是指图中与该顶点相邻接的顶点数。若无向图 G 中的顶点数为 n、边数为 e，则所有顶点的度数之和为__(59)__。(2011 年下半年试题 59)

(59) A. n*e　　B. $n+e$　　C. $2n$　　D. $2e$

解析： 在无向图中，一条边与两个顶点相连，边数为 e 的无向图所有顶点的度数之和为 $2e$。

答案： A

例 4　设一个包含 n 个顶点、e 条弧的简单有向图采用邻接矩阵存储结构(即矩阵元素 $A[i][j]$分别等于 1 或 0，表示顶点 i 与顶点 j 之间有弧或无弧)，则该矩阵的非零元素数目为__(61)__。(2015 年上半年试题 61)

(61) A. e　　B. $2e$　　C. $n-e$　　D. $n+e$

解析： 用邻接矩阵存储有向图，图中每一条弧对应矩阵一个非零元素，题目中提到共有 e 条弧，所以共 e 个非零元素。

答案： A

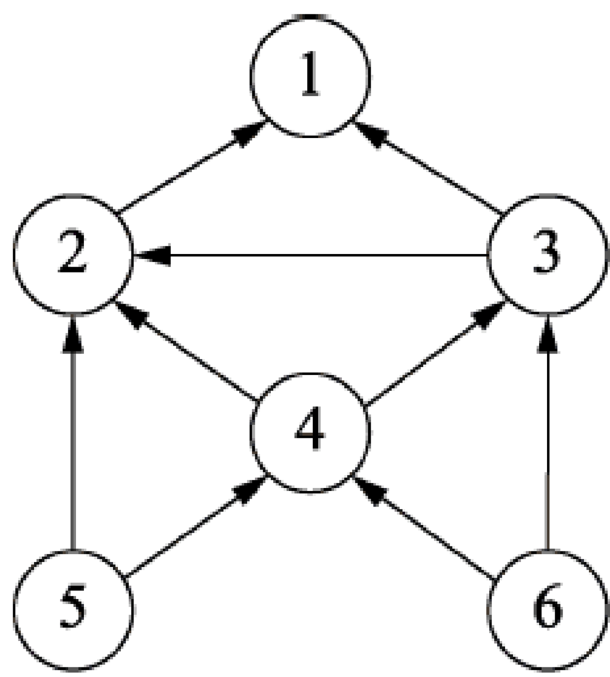
例 5　设一个包含 N 个顶点、E 条边的简单无向图采用邻接矩阵存储结构(矩阵元素 $A[i][j]$等于 1 或 0 分别表示顶点 i 与顶点 j 之间有或无边)，则该矩阵中的非零元素数据为(60)。(2011 年上半年试题 60)

(60) A. N　　B. E　　C. $2E$　　D. $N+E$

解析： 邻接矩阵是一个用来存放顶点间关系(边或弧)数据的二维数组，如果顶点间存在边，则用 1 表示，用 0 表示不存在的边。在无向图中，邻接矩阵中的内容是对称的，如果顶点 A 和顶点 B 之间存在公共边，则表示顶点 A 可以到达顶点 B，顶点 B 也可到达顶点 A。如果简单无向图有 E 条边，则邻接矩阵中非零元素数据有 $2E$ 个。

答案： C

例 6　__(59)__是右图的合法拓扑序列。(2010 年下半年试题 59)



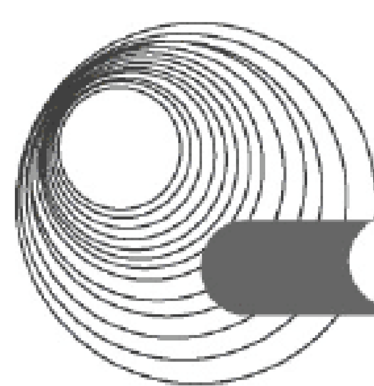
(59) A. 6 5 4 3 2 1　　B. 1 2 3 4 5 6　　C. 5 6 3 4 2 1　　D. 5 6 4 2 1 3

解析： 拓扑排序是将 AOV 网中所有顶点排成一个线性序列的过程。对 AOV 网进行拓扑排序的方法如下。

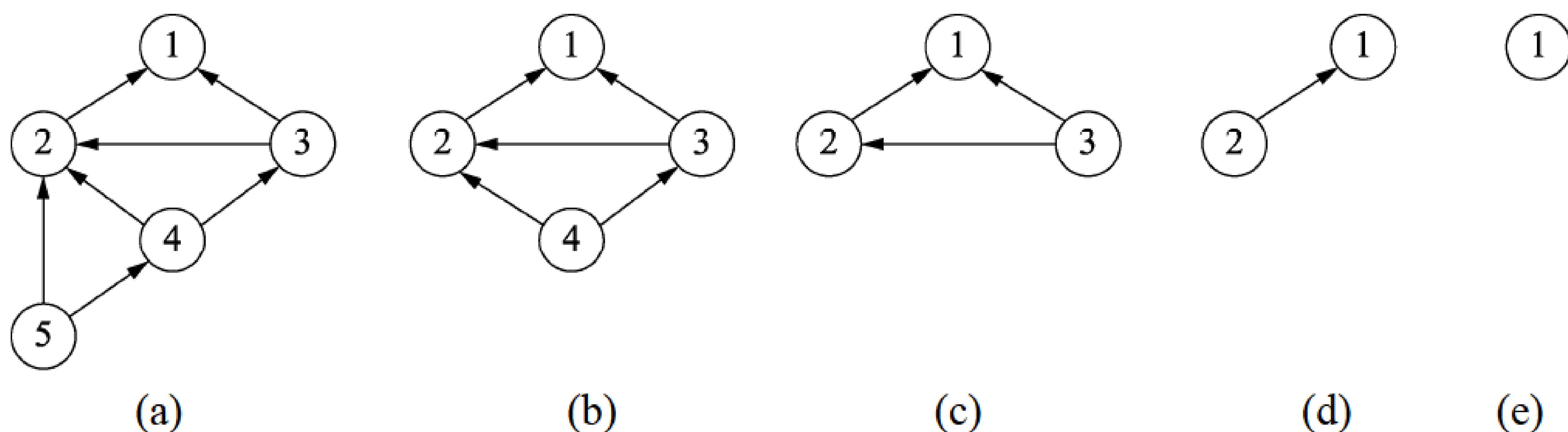
① 在 AOV 网中选择一个入度为 0 的顶点，输出它。

② 从网中删除该顶点及其与该顶点有关的所有边。

③ 重复上述两步，直至网中不存在入度为 0 的顶点为止。



本题的拓扑排序过程如下图所示。



得到的拓扑序列为 6 5 4 3 2 1。

答案：A

例 7　以下关于图的遍历叙述中，正确的是 (61) 。(2016 年上半年试题 61)

(61) A．图的遍历是从给定的源点出发对每一个顶点仅访问一次的过程

B．图的深度优先遍历方法不适用于无向图

C．使用队列对图进行广度优先遍历

D．图中有回路时则无法进行遍历

解析：广度优先。当一个节点被加入队列时，要标记为已遍历，遍历过程中，对于队列第一个元素，遍历其所有能够一步达到的节点，如果是标记未遍历的，将其加入队列，从第一个元素出发所有能一步直接达到的节点遍历结束后将这个元素出列。

深度优先。当遍历到某个节点 A 时，如果是标记未遍历，则将其入栈，遍历它能够一步直接达到的节点，如果是标记未遍历，则将其入栈且标记为已遍历，然后对其进行类似 A 的操作；否则找能够一步直接达到的节点进行类似操作。直到所有能够一步直接达到的节点都已遍历，将 A 出栈。

答案：C

例 8　下面关于图(网)的叙述，正确的是(58)。(2009 年上半年试题 58)

(58) A．连通无向网的最小生成树中，顶点数恰好比边数多 1

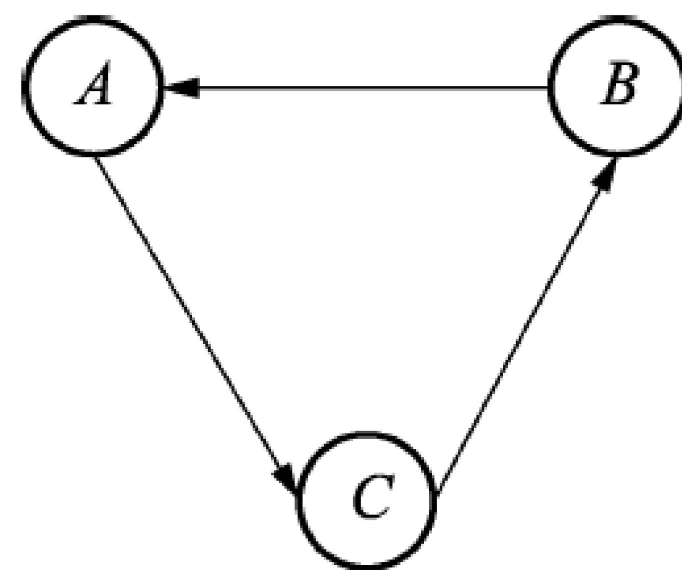
B．若有向图是强连通的，则其边数至少是顶点数的 2 倍

C．可以采用 AOV 网估算工程的工期

D．关键路径是 AOE 网中源点至汇点的最短路径

解析：生成树即极小连通子图，包含图的所有 n 个节点，但只含图的 n-1 条边。在生成树中添加一条边之后，必定会形成回路或环。所以 A 正确。

在有向图 G 中，如果对于每一对 V_i，$V_j \in V$，$V_i \neq V_j$，从 V_i～V_j 和从 V_j～V_i 都存在路径，则称 G 是强连通图。右图为强连通图，但其边数等于顶点数。



AOV 网：用顶点表示活动，用弧表示活动间的优先关系的有向图。

AOE 网：用顶点表示事件，弧表示活动，权表示活动持续的时间。通常 AOE 网用来估算工程的完成时间。

关键路径：由于在 AOE 网中有些活动可以并行地进行，所以完成工程的最短时间是从开始点到完成点的最长路径的长度(指路径上各活动持续时间之和，不是路径上弧的数目)，

最长的路径叫做关键路径。

答案：A

3.4.3　同步练习

1. ______的邻接矩阵是一个对称矩阵。

A．无向图　　B．AOV 网　　C．AOE 网　　D．有向图

2. 具有 n 个顶点、e 条边的图采用邻接表存储结构，进行深度优先遍历和广度优先遍历运算的时间复杂度均为______。

A．$O(n^2)$　　B．$O(e^2)$　　C．$O(n\times e)$　　D．$O(n+e)$

3. 设一个包含 N 个顶点、E 条边的简单有向图采用邻接矩阵存储结构(矩阵元素 $A[i][j]$ 等于 1 或 0 分别表示顶点 i 与顶点 j 之间有或无弧)，则该矩阵的元素数目为(1)，其中非零元素数目为(2)。

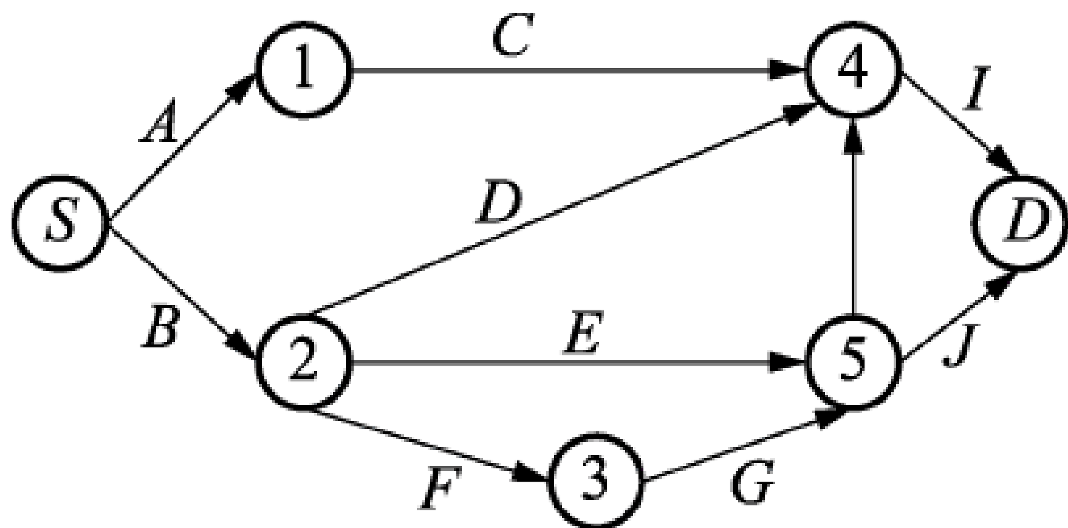
(1) A．E^2　　B．N^2　　C．N^2-E^2　　D．N^2+E^2

(2) A．N　　B．$N+E$　　C．E　　D．$N-E$

4. 拓扑排序是指有向图中的所有顶点排成一个线性序列的过程，若在有向图中从顶点 v_i 到 v_j 有一条路径，则在该线性序列中，顶点 v_i 必然在顶点 v_j 之前。因此，若不能得到全部顶点的拓扑排序序列，则说明该有向图一定______。

A．包含回路　　B．是强连通图　　C．是完全图　　D．是有向树

5. 某工程计划如下图所示，各个作业所需的天数如下表所示，设该工程从第 0 天开工，则该工程的最短工期是 (1) 天，作业 J 最迟应在第 (2) 天开工。

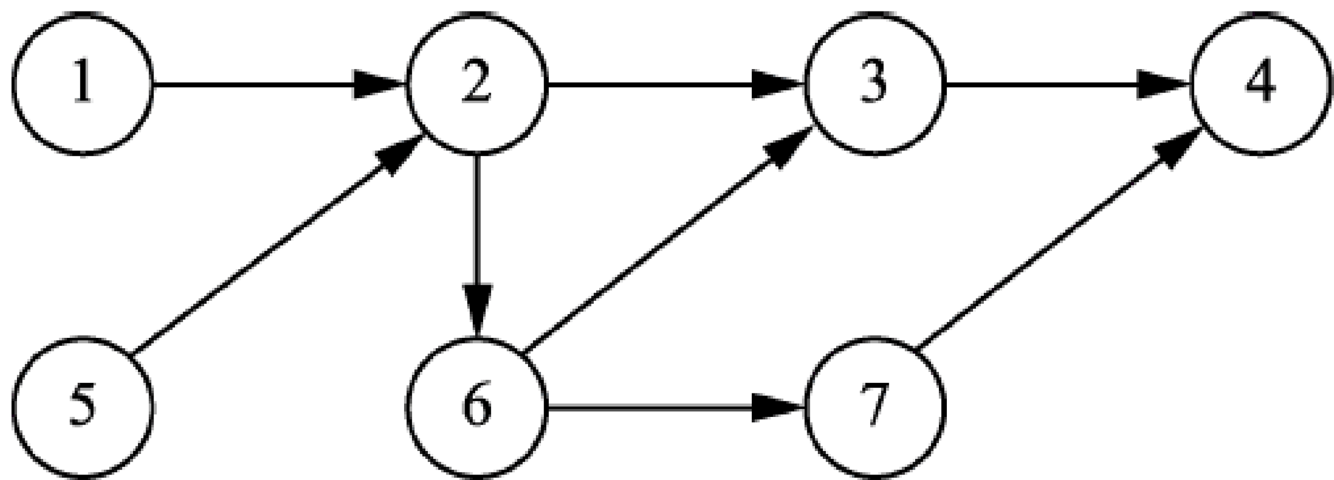


作业	A	B	C	D	E	F	G	H	I	J
所需天数	7	6	8	10	7	3	2	4	3	7

(1) A．17　　B．18　　C．19　　D．20

(2) A．11　　B．13　　C．14　　D．16

6. 拓扑序列是无环有向图中所有顶点的一个线性序列，图中任意路径中的各个顶点在该图的拓扑序列中保持先后关系，______为下图所示有向图的一个拓扑序列。



A. 1 2 3 4 5 6 7　B. 1 5 2 6 3 7 4　C. 5 1 2 6 3 4 7　D. 5 1 2 3 7 6 4

7. 一个含有 n 个顶点、e 条边的简单无向图，在其邻接矩阵存储结构中共有个零元素。

A. e　B. $2e$　C. n^2-e　D. n^2-2e

8. 若采用邻接矩阵来存储简单有向图，则其某一个顶点 i 的入度等于该矩阵。

A. 第 i 行中值为 1 的元素个数

B. 所有值为 1 的元素总数

C. 第 i 行及第 i 列中值为 1 的元素总个数

D. 第 i 列中值为 1 的元素个数

3.4.4 同步练习参考答案

1. A　　2. D

3. (1) B　(2) C　　4. A

5. (1) D　(2) B　　6. B

7. D　　8. D

3.5 查　找

3.5.1 考点辅导

3.5.1.1 查找的基本概念

1. 基本概念

查找是一种常用的基本运算。查找表是指由同一类型的数据元素构成的集合。

- 静态查找表。对查找表经常要进行的两种操作是查询和检索。
- 动态查找表。对查找表经常要进行的操作是插入和删除。
- 关键字。数据元素的某个数据项的值，用它来识别这个数据元素。
- 主关键字。能唯一标识一个数据元素的关键字。
- 次关键字。能标识多个数据元素的关键字。
- 查找。根据给定的某个值，在查找表中确定是否存在一个其关键字等于给定值的记录或数据元素的过程称为查找。

2. 查找操作的性能分析

通常以“其关键字和给定值进行过比较的记录个数的平均值”作为衡量查找算法好坏的依据。

平均查找长度：为确定记录在查找表中的位置，须与给定关键字值进行比较的次数的数学期望称为查找算法在查找成功时的平均查找长度。

对于含有 n 个记录的表，平均查找长度 ASL 定义为

$$\mathrm{ASL}=\sum_{i=1}^{n}p_ic_i$$

式中，p_i 为对表中第 i 个记录进行查找的概率，且 $\sum_{i=1}^{n}p_i=1$。一般情况下，均认为查找每个记录的概率是相等的，即 $p_i=1/n$。

3.5.1.2　静态查找表

1. 顺序查找

顺序查找的基本思想是：从表的一端开始，逐个进行记录的关键字和给定值的比较，若找到一个记录的关键字与给定值相等，则查找成功；若整个表中的记录均比较过，仍未找到关键字等于给定值的记录，则查找失败。

顺序查找的性能分析如下。

一般情况下，$c_i=n-i+1$，因此在等概率情况下，顺序查找成功的平均查找长度为

$$\mathrm{ASL}_{\mathrm{ss}}=\sum_{i=1}^{n}p_ic_i=\frac{1}{n}\sum_{i=1}^{n}(n-i+1)=\frac{n+1}{2}$$

也就是说，成功查找的平均次数约为表长的一半。与其他方法相比，顺序查找方法在 n 值较大时，其平均查找长度较大，查找效率较低。但这种方法也有优点，就是算法简单且适应面广，对查找表的结构没有要求，无论记录是否按关键字有序排列均可使用。

2. 折半查找

折半查找的基本思想是：设查找表的元素存储在一维数组 $r[1\cdots n]$ 中，那么在表中的元素已经按关键字递增(或递减)的方式排序的情况下，可进行折半查找。其方法是：首先将待查的 key 值与表 r 中间位置上(下标为 mid)记录的关键字进行比较，若相等则查找成功；若 key> r[mid].key，则说明待查记录只可能在后半个子表 $r[\mathrm{mid}+1\cdots n]$ 中，下一步应在后半个子表中再进行折半查找；若 key<r[mid].key，说明待查记录只可能在前半个子表 $r[1\cdots \mathrm{mid}-1]$ 中，下一步应在 r 的前半个子表中进行折半查找。这样通过逐步缩小范围，直到查找成功或子表为空时失败为止。

折半查找的性能分析：折半查找的过程可以用一棵二叉树描述，不妨设节点总数为 $n=2^h-1$，则判定树是深度为 $h=\log_2(n+1)$ 的满二叉树。在等概率情况下，折半查找的平均查找长度为 $\mathrm{ASL}_{\mathrm{bs}}=\sum_{i=1}^{n}p_ic_i=\frac{1}{n}\sum_{j=1}^{n}j\times 2^{j-1}=\frac{n+1}{n}\log_2(n+1)-1$。

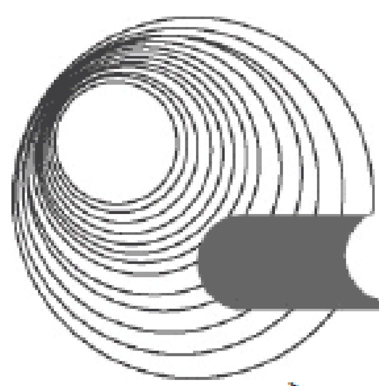
当 n 值较大时，$\mathrm{ASL}_{\mathrm{bs}}\approx\log_2(n+1)-1$。

折半查找比顺序查找的效率高，但它要求查找表进行顺序存储并且按关键字有序排列，因此对表进行元素的插入和删除时，需要移动大量的元素，所以折半查找适用于表不易变动，且又经常进行查找的情况。

3. 分块查找

分块查找又称为索引顺序查找，是对顺序查找方法的一种改进，其性能介于顺序查找和二分查找之间。

分块查找的基本思想是：在分块查找过程中，首先把表分成若干块，每一块中的关键



字不一定有序，但块之间是有序的，即后一块中所有记录的关键字均大于前一个块中最大的关键字。此外，还建立了一个索引表，索引表按关键字有序。所以分块查找的过程分为两步：第一步在索引表中确定待查记录所在的块；第二步在块内顺序查找。

3.5.1.3 动态查找表

1. 二叉排序树

二叉排序树又称为二叉查找树，它或者是一棵空树，或者是满足以下性质的二叉树。

(1) 若它的左子树非空，则左子树上所有节点的值均小于根节点的值。

(2) 若它的右子树非空，则右子树上所有节点的值均大于根节点的值。

(3) 左、右子树本身就是两棵二叉排序树。

二叉排序树的查找过程是：若二叉排序树非空，则将给定值与根节点的关键字值相比较，若相等，则查找成功；若不等，则当根节点的关键字值大于给定值时，到根的左子树中进行查找；否则到根的右子树中进行查找。若找到，则查找过程是走了一条从树根到所找到节点的路径；否则查找过程终止于一棵空树。

二叉排序树中插入节点的操作：每读入一个元素，建立一个新节点，若二叉树非空，则将新节点的值与根节点的值相比较，如果小于根节点的值，则插入到左子树中，否则插入到右子树中；若二叉排序树为空，则新节点作为二叉排序树的根节点。

二叉排序树中删除节点的操作：在二叉树中删除一个节点，不能把以该节点为根的子树都删除，只能删除这个节点并仍旧保持二叉排序树的特性，也就是说，删除二叉排序树上一个节点相当于删除有序数列中的一个元素。假设二叉排序树上的被删除节点为*p(p 指针指向被删除节点)，*f 为其双亲节点，则删除节点*p 的过程可分为以下 3 种情况。

① 若*p 节点为叶子节点，即 p→lchild 及 p→rchild 均为空，则由于删去叶子节点后不破坏整棵树的结构，因此只需修改*p 节点的双亲节点*f 的相应指针即可，即 f→lchild(或 f→rchild)=NULL。

② 若*p 节点只有左子树或者只有右子树，此时只要将*p 的左子树或右子树接成其双亲节点*f 左子树或右子树，即令 f→lchild(或 f→rchild)=p→lchild，或 f→lchild(或 f→rchild)=p→rchild。

③ 若*p 节点的左、右子树均不空，则不能像上面那样简单处理，删除*p 节点时应将*p 的左子树、右子树连接到适当的位置，并保持二叉排序树的特性。可采用以下两种方法进行处理：一是令*p 的左子树为*f 的左(或右)子树，而将*p 的右子树下接到*p 的中序遍历的直接前驱节点*s 的右孩子指针上；二是用*p 的中序直接前驱(或后继)节点*s 代替*p 节点，然后删除*s 节点。

2. 平衡二叉树

平衡二叉树又称为 AVL 树，它或者是一棵空树，或者是具有下列性质的二叉树：它的左、右子树都是平衡二叉树，且左子树和右子树的深度之差的绝对值不超过 1；若将二叉树节点的平衡因子定义为该节点的左子树的深度减去其右子树的深度，则平衡树上所有节点的平衡因子只可能是-1、0 和 1；只要树上有一个节点的平衡因子的绝对值大于 1，则该二叉树就是不平衡的。

平衡二叉树上的插入操作。失去平衡后进行调整的规律可归纳为 4 种情况：①单向右

旋平衡处理；②单向左旋平衡处理；③双向旋转(从左到右)平衡处理；④双向旋转(从右到左)平衡处理。

平衡二叉树上的删除操作：若删除节点的两个子树都不为空，就用该节点左子树上的中序遍历的最后一个节点(或其右子树上的第一个节点)替换该节点，将情况转化为待删除的节点只有一个子树后再进行处理。当一个节点被删除后，从被删节点到树根的路径上所有节点的平衡因子都需要更新，对于每一个位于该路径上的平衡因子为±2 的节点来说，都要进行平衡处理。

3. B-树

B-树的定义：一棵 m 阶的 B-树，或为空树，或为满足下列特性的 m 叉树。

(1) 树中每个节点至多有 m 棵子树。

(2) 若根节点不是叶子节点，则至少有两棵子树。

(3) 除根之外的所有非终端节点至少有 $\left\lceil \frac{m}{2} \right\rceil$ 棵子树。

(4) 所有的非终端节点中包含下列数据信息，即

$$(n, A_0, K_1, A_1, K_2, A_2, \cdots, K_n, A_n)$$

式中，K_i (i=1, 2, …, n)为关键字，且 $K_i < K_{i+1}$ (i=1, 2, …, n−1)；A_i (i=1, 2, …, n)为指向子树根节点的指针，且指针 A_{i-1} 所指子树中所有节点的关键字均小于 K_i (i=1, 2, …, n)，A_n 所指子树中所有节点的关键字均大于 K_n，$n\left(\left\lceil \frac{m}{2} \right\rceil - 1 \leqslant n \leqslant m-1\right)$为节点中关键字的个数。

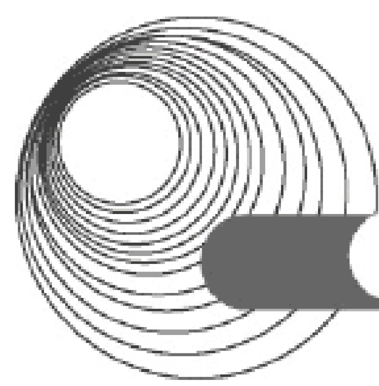
(5) 所有的叶子节点都出现在同一层次上，并且不带信息(可以看作外部节点或查找失败的节点，实际上这些节点不存在，指向这些节点的指针为空)。

B-树上进行查找的过程是：首先在根节点所包含的关键字中查找给定的关键字，若找到则成功返回；否则确定待查找的关键字所在的子树并继续进行查找，直到查找成功或查找失败(指针为空)时为止。

B-树上的插入和删除运算较为复杂，因为要保证运算后节点中关键字的个数大于等于 $\left\lceil \frac{m}{2} \right\rceil - 1$，因此涉及节点的“分裂”及“合并”问题。

在 B-树中插入一个关键字时，不是在树中加一个叶子节点，而是首先在低层的某个终端节点添加一个关键字，若该节点中关键字的个数不超过 m−1，则完成插入；否则，要进行节点的“分裂”处理。“分裂”就是把节点中处于中间位置上的关键字取出来插入到其父节点中，并以该关键字为分界线，把原节点分成两个节点，“分裂”过程可能会一直持续到树根。

在 B-树中删除一个节点时，首先找到关键字所在的节点，若该节点在含有信息的最后一层，且其中关键字的数目不少于 $\left\lceil \frac{m}{2} \right\rceil - 1$，则完成删除；否则需进行节点的“合并”运算。若待删除的关键字所在节点不在含有信息的最后一层上，则将该关键字用其在 B-树中的后继替代，然后再删除其后继元素，即将需要处理的情况统一转化为在含有信息的最后一层再进行删除运算。



3.5.1.4 哈希表及其查找

1. 定义

根据设定的哈希函数和处理冲突的方法，将一组关键字映射到一个有限的、连续的地址集(区间)上，并以关键字在地址集中的“像”作为记录在表中的存储位置，这种表称为哈希表，这一映射过程称为哈希造表或散列，所得的存储位置称为哈希地址或散列地址。

对于哈希表，主要考虑两个问题：一是如何构造哈希函数；二是如何解决冲突。

2. 哈希函数的构造方法

常用的哈希函数构造方法有直接定址法、数字分析法、平方取中法、折叠法、随机数法和除留余数法等。

3. 处理冲突的方法

常见的处理冲突的方法有开放地址法、链地址法、再哈希法、建立一个公共溢出区。

4. 哈希表的查找及其性能分析

从哈希表的查找过程可知以下两点。

(1) 虽然哈希表在关键字与记录的存储位置之间建立了直接映像，但由于冲突的产生，使得哈希表的查找过程仍然是一个给定值和关键字进行比较的过程，因此，仍须以平均查找长度衡量哈希表的查找效率。

(2) 查找过程中须与给定值进行比较的关键字的个数取决于哈希函数、处理冲突的方法和哈希表的装填因子 3 个因素。哈希表的装填因子定义为

$$\alpha = \frac{\text{表中装入的记录数}}{\text{哈希表的长度}}$$

式中，α 表示哈希表的装满程度。直观地看，α 越小，发生冲突的可能性就越小；反之，α 越大，表中已填入的记录越多，再填记录时，发生冲突的可能性就越大，则查找时，给定值需与之进行比较的关键字的个数也就越多。

3.5.2 典型例题分析

例 1 以下关于哈希(Hash，散列)查找的叙述中，正确的是 (65) 。(2013 年上半年试题 65)

(65) A. 哈希函数应尽可能复杂些，以消除冲突

B. 构造哈希函数时应尽量使关键字的所有组成部分都能起作用

C. 进行哈希查找时，不再需要与查找表中的元素进行比较

D. 在哈希表中只能添加元素不能删除元素

解析：哈希表中的元素是由哈希函数确定的。将数据元素的关键字 K 作为自变量，通过一定的函数关系(称为哈希函数)计算出的值即为该元素的存储地址。所以在构造哈希函数时应尽量使关键字的所有组成部分起作用。

答案：B

例 2 在 55 个互异元素构成的有序表 $A[1..55]$中进行折半查找(或二分查找，向下取整)。若需要找的元素等于 $A[19]$，则在查找过程中参与比较的元素依次为 (60) 、$A[19]$。(2015 年下

半年试题 60)

(60) A．$A[28]$、$A[30]$、$A[15]$、$A[20]$

B．$A[28]$、$A[14]$、$A[21]$、$A[17]$

C．$A[28]$、$A[15]$、$A[22]$、$A[18]$

D．$A[28]$、$A[18]$、$A[22]$、$A[20]$

解析：折半查找时，下标计算过程为(注：key 的值与 $A[19]$相同)：

(1) mid=[(1+55) /2]=28，把 $A[28]$与 key 的值比较后，缩小查找范围为：$A[1]$~$A[27]$。

(2) mid=[(1+27)/2]=14，把 $A[14]$与 key 的值比较后，缩小查找范围为：$A[15]$~$A[27]$。

(3) mid=[(15+27)/2]=21，把 $A[21]$与 key 的值比较后，缩小查找范围为：$A[15]$~$A[20]$。

(4) mid=[(15+20)/2]=17，把 $A[17]$与 key 的值比较后，缩小查找范围为：$A[18]$~$A[20]$。

(5) mid=[(18+20)/2]=19，把 $A[19]$与 key 的值比较后，发现值相等，找到目标。

答案：B

例 3　对 n 个元素的有序表 $A[1..n]$进行顺序查找，其成功查找的平均查找长度(即在查找表中找到指定关键码的元素时，所进行比较的表中元素个数的期望值)为<u>(58)</u>。(2011 年上半年试题 58)

(58) A．n　　B．$(n+1)/2$　　C．$\log_2 n$　　D．n^2

解析：在等概率情况下，顺序查找的期望值为 $\sum_{i=1}^{n} P_i C_i = \frac{1}{n}\sum_{i=1}^{n}(n-i+1) = \frac{n+1}{2}$。

答案：B

例 4　对于关键字序列(26, 25, 72, 38, 8, 18, 59)，采用散列函数 $H(\text{Key})$=Key mod 13 构造散列表(哈希表)。若采用线性探测的开放定址法解决冲突(顺序地探查可用存储单元)，则关键字 59 所在散列表中的地址为<u>(61)</u>。(2011 年上半年试题 61)

(61) A．6　　B．7　　C．8　　D．9

解析：26 mod 13 =0，25 mod 13 = 12，72 mod 13=7，38 mod 13=12，8 mod 13=8，18 mod 13=5，59 mod 13=7。

元素 26 的哈希地址为 0；25 的哈希地址为 12；72 的哈希地址为 7；对于元素 38，其哈希地址是 12，但是该地址已经存入元素 25，因此试探哈希地址 13，没有冲突，因此将元素 38 存入哈希地址为 13 的单元；元素 8 的哈希地址为 8；元素 18 的哈希地址为 5；元素 59 的哈希地址为 7，但是该地址已经存入元素 72，再试探哈希地址 8，发生冲突，地址 8 被元素 8 占用，再试探哈希地址 9，没有冲突，因此将元素 59 存入哈希地址为 9 的单元。

答案：D

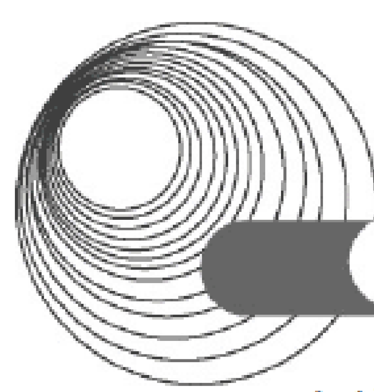
例 5　某一维数组中依次存放了数据元素 15、23、38、47、55、62、88、95、102、123，采用折半(二分)法查找元素 95 时，依次与<u>(60)</u>进行了比较。(2010 年下半年试题 60)

(60) A．62, 88, 95　　B．62, 95　　C．55, 88, 95　　D．55, 95

解析：折半查找要求线性表是有序排列的，本题中的数据已按升序排好，设元素保存在一维数组 $r[\text{low..high}]$。进行折半查找的具体方式如下。

首先确定数组的中间位置 mid，mid=(low+high)/2。

将待查的 k 值与 r 中间位置上的元素进行比较，如果相等，则查找成功；若 $k<r[\text{mid}]$，则待查的 k 值必定在子区间 $r[\text{low..mid}-1]$中，因此新的区间为 $r[\text{low..mid}-1]$；如果 $k>r[\text{mid}]$，



则待查的 k 值必定在子区间 r[mid+1..high]中，因此新的区间为 r[mid+1..high]。

然后在新的区间进行查找。

本题中第一次查找时，low=1，high=10，mid=5，r[mid]=55，95>r[mid]；将 low 设置为 mid+1=6，high=10，则 mid=8，r[mid]=95，查找成功。

答案：D

例 6 某哈希表(散列表)的长度为 n，设散列函数为 H(Key)=Key mod p，采用线性探测法解决冲突。以下关于 p 值的叙述中，正确的是 (61) 。(2013 年下半年试题 61)

(61) A．p 的值一般为不大于 n 且最接近 n 的质数

B．p 的值一般为大于 n 的任意整数

C．p 的值必须为小于 n 的合数

D．p 的值必须等于 n

解析：如果参数 p 是合数的话，那么 Key 相对于 p 的模得到的散列值会有很多是相同的。所以，p 一般取质数，如果 p 的值大于散列表的长度，散列函数得到的散列地址将和 Key 的范围大小相同，那么散列函数也就没有意义了，所以答案选 A。

答案：A

例 7 在字符串的 KMP 模式匹配算法中，需先求解模式串的 next 函数值，其定义如下式所示，j 表示模式串中字符的序号(从 1 开始)。若模式串 p 为"abaac"，则其 next 函数值为 (60) (2014 年下半年试题 60)。

$$\text{next}[j]=\begin{cases}0 & j=1\\ \max\{k\,|\,1<k<j,'p_1p_2Lp_{k-1}'='p_{j-k+1}p_{j-k+2}Lp_{j-1}'\} & \\ 1 & \text{其他情况}\end{cases}$$

(60) A．01234　　B．01122　　C．01211　　D．01111

解析：根据公式依次推导即可。

答案：B

例 8 实现二分查找(折半查找)时，要求查找表 (61) 。(2014 年上半年试题 61)

(61) A．顺序存储，关键码无序排列

B．顺序存储，关键码有序排列

C．双向链表存储，关键码无序排列

D．双向链表存储，关键码有序排列

解析：二分查找又称为折半查找，优点是比较次数少，查找速度快，平均性能好；其缺点是要求待查表为有序表，且插入和删除困难。因此，折半查找方法适用于不经常变动而查找频繁的有序列表。

二分查找算法要求：①必须采用顺序存储结构；②必须按关键字大小有序排列。

答案：B

例 9 对某有序顺序表进行折半查找时， (60) 不可能构成查找过程中关键字的比较序列。(2015 年下半年试题 60)

(60) A．45,10,30,18,25　　B．45,30,18,25,10

C.10,45,18,30,25　　D．10,18,25,30,45

解析：折半查找要求待查表为有序表。首先，假设表中元素是按升序排列，将表中间

位置记录的关键字与查找关键字比较，如果两者相等，则查找成功；否则利用中间位置记录将表分成前、后两个子表，如果中间位置记录的关键字大于查找关键字，则进一步查找前一子表；否则进一步查找后一子表。重复以上过程，直至找到满足条件的记录，使查找成功，或直到子表不存在为止，此时查找不成功。

对于选项 B，待查元素的关键字第一次与 45 比较，第 2 次与 30 比较，说明关键字比 45 小；第 3 次与 18 比较，说明关键字比 30 小；第 4 次与 25 比较，说明关键字比 18 大。可见，第 5 次不可能与 10 比较，至少比较的值要比 18 大。

答案：B

3.5.3　同步练习

1.　将一个无序序列中的元素依次插入到一棵______，并进行中序遍历，可得到一个有序序列。

A．完全二叉树　B．最小生成树　C．二叉排序树　D．最优二叉树

2.　某一维数组中依次存放了数据元素 12、23、30、38、41、52、54、76、85，在用折半(二分)查找方法(向上取整)查找元素 54 时，所经历“比较”运算的数据元素依次为______。

A．41、52、54　　B．41、76、54

C．41、76、52、54　　D．41、30、76、54

3．已知一个线性表(16、25、35、43、51、62、87、93)，采用散列函数 H(Key)=Key mod 7 将元素散列到表长为 9 的散列表中。若采用线性探测的开放定址法解决冲突(顺序地探查可用存储单元)，则构造的哈希表为__(1)__，在该散列表上进行等概率成功查找的平均查找长度为__(2)__(确定为记录在查找表中的位置，需和给定关键字值进行比较的次数的期望值称为查找算法在查找成功时的平均查找长度)。

(1)

A.

0	1	2	3	4	5	6	7	8
35	43	16	51	25		62	87	93

B.

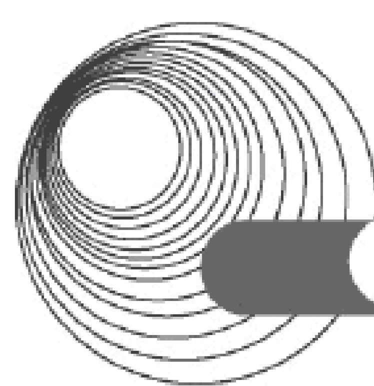
0	1	2	3	4	5	6	7	8
35	43	16	93	25	51	62	87	

C.

0	1	2	3	4	5	6	7	8
35	43	16	51	25	87	62	93	

D.

0	1	2	3	4	5	6	7	8
35	43	16	51	25	87	62		93



(2) A．(5×1+2+3+6) / 8　　B．(5×1+2+3+6) / 9
C．(8×1) / 8　　D．(8×1) / 9

4．已知一个线性表(38, 25, 74, 63, 52, 48)，假定采用散列函数 H(key)=key%7 计算散列地址，并将散列存储在散列表 A[0..6]中，若采用线性探测方法解决冲突，则在该散列表上进行等概率成功查找的平均查找长度为______。

A．1.5　　B．1.7　　C．2.0　　D．2.3

5．设顺序存储的某线性表共有 123 个元素，按分块查找的要求等分为 3 块。若对索引表采用顺序查找方法来确定子块，且在确定的子块中也采用顺序查找方法，则在等概率的情况下，分块查找的平均查找长度为______。

A．21　　B．23　　C．41　　D．62

6．类比二分搜索算法，设计 k 分搜索算法(k 为大于 2 的整数)如下：首先检查 n/k 处(n 为被搜索集合的元素个数)的元素是否等于要搜索的值，然后检查 $2n/k$ 处的元素，……，这样，或者找到要搜索的元素，或者把集合缩小到原来的 $1/k$；如果未找到要搜索的元素，则继续在得到的集合上进行 k 分搜索；如此进行，直至找到要搜索的元素或搜索失败。此 k 分搜索算法在最坏情况下搜索成功的时间复杂度为 (1) ，在最好情况下搜索失败的时间复杂度为 (2) 。

(1)～(2) A．$O(\log_2 n)$　　B．$O(n\log_2 n)$
C．$O(\log_k n)$　　D．$O(n\log_k n)$

3.5.4 同步练习参考答案

1．C　　2．B
3．(1) C　(2) A　　4．C
5．B　　6．(1) C　(2) C

3.6 排　　序

3.6.1 考点辅导

3.6.1.1 排序的基本概念及运算

排序：假设含 n 个记录的文件内容为$\{R_1, R_2, \cdots, R_n\}$，其相应的关键字分别为$\{K_1, K_2, \cdots, K_n\}$。经过排序确定一种排列：$R_{i1}$，$R_{i2}$，…，$R_{in}$，使得它们的关键字满足关系 $K_{i1} \leqslant K_{i2} \leqslant \cdots \leqslant K_{in}$(或 $K_{i1} \geqslant K_{i2} \geqslant \cdots \geqslant K_{in}$)，这样的运算称为排序。

内部排序：指待排序记录全部存放在内存中排序的过程。

外部排序：指待排序记录的数量很大，以至内存不能容纳全部记录，在排序过程中尚需对外存进行访问的过程。

3.6.1.2　简单排序

下面介绍几种简单排序方法。

(1)　直接插入排序。在插入第 i 个记录时，$R_1, R_2, \cdots, R_{i-1}$ 已经排好序，这时将关键字 k_i 依次与关键字 $k_{i-1}, k_{i-2}, \cdots, k_1$ 进行比较，从而找到应该插入的位置，然后将 k_i 插入，插入位置及其后的记录依次向后移动。

(2)　冒泡排序。首先将第一个记录的关键字和第二个记录的关键字进行比较，若为逆序，则交换两个记录的值，然后比较第二个记录和第三个记录的关键字，以此类推，直至第 n−1 个记录和第 n 个记录的关键字进行过比较为止。上述过程称为第一趟冒泡排序，其结果是关键字最大的记录被安置到第 n 个记录的位置上，然后进行第二趟冒泡排序，对前 n−1 个记录进行同样的操作，其结果是关键字次大的记录被安置到第 n−1 个记录的位置上，当进行完第 n−1 趟时，所有记录有序排列。

(3)　简单选择排序。通过 n−1 次关键字之间的比较，从 n−i+1 个记录中选出关键字最小的记录，并和第 i 个记录进行交换，当 i 等于 n 时所有记录有序排列。

3.6.1.3　希尔排序

希尔排序又称为缩小增量排序，是对直接插入排序方法的改进。

希尔排序的基本思想是：先将整个待排记录序列分割成若干个子序列，然后分别进行直接插入排序，待整个序列中的记录基本有序时，再对全体记录进行一次直接插入排序。具体做法是：先取定一个小于 n 的整数 d_1 作为第一个增量，把文件的全部记录分成 d_1 个组，将所有距离为 d_1 倍数的记录放在同一个组中，在各组内进行直接插入排序；然后取第二个增量 $d_2 < d_1$，重复上述分组和排序工作，以此类推，直至所取的增量 $d_i = 1(d_i < d_{i-1} < \cdots < d_2 < d_1)$，即所有记录放在同一组进行直接插入排序为止。

3.6.1.4　快速排序

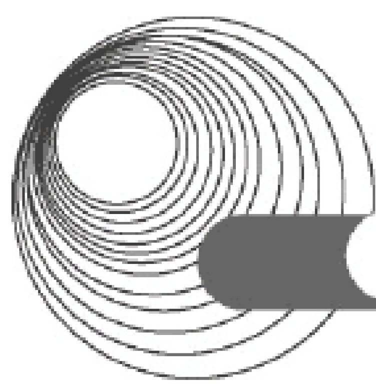
快速排序的基本思想是：通过一趟排序将待排的记录分割为独立的两部分，其中一部分记录的关键字均比另一部分记录的关键字小，然后再分别对这两部分记录继续进行排序，以达到整个序列有序。

具体做法是：附设两个指针 low 和 high，它们的初值分别指向文件的第一个记录和最后一个记录。设枢轴记录的关键字为 Pivotkey，则首先从 high 所指位置起向前搜索，找到第一个关键字小于 Pivotkey 的记录并与枢轴记录互相交换，然后从 low 所指位置起向后搜索，找到第一个关键字大于 Pivotkey 的记录并与枢轴记录相互交换，重复这两步直至 low=high 为止。

在所有同数量级 $(O(n\log_2 n))$ 的排序方法中，快速排序被认为是平均性能最好的一种，但是，若初始记录序列按关键字有序或基本有序时，快速排序将退化为冒泡排序，此时算法的时间复杂度为 $O(n^2)$。

3.6.1.5　堆排序

对于 n 个元素的关键字序列 K_1，K_2，⋯，K_n，当且仅当所有关键字都满足下列性质时称其为堆，即



$$\begin{cases} K_i \leqslant K_{2i} \\ K_i \leqslant K_{2i+1} \end{cases} \text{或} \begin{cases} K_i \geqslant K_{2i} \\ K_i \geqslant K_{2i+1} \end{cases} \quad i=1,2,\cdots,\left[\frac{n}{2}\right]$$

若堆顶为最小元素，则称为小根堆；若堆顶为最大元素，则称为大根堆。

堆排序的基本思想是：对一组待排序记录的关键字，首先把它们按堆的定义排成一个堆序列，从而输出堆顶的最小关键字(对于小根堆而言)，然后将剩余的关键字再调整成新堆，便得到次小的关键字，如此反复进行，直到全部关键字排成有序序列为止。

对于记录数较少的文件来说，堆排序的优越性并不明显，但对大量的记录来说堆排序是很有效的。堆排序的整个算法时间是由建立堆和不断调整堆这两部分时间代价构成的，堆排序算法的时间复杂度为$O(n\log_2 n)$。此外，堆排序只需要一个记录大小的辅助空间。但是堆排序是一种不稳定的排序方法。

3.6.1.6 归并排序

归并是将两个或两个以上的有序文件合并成为一个新的有序文件。

归并排序是把一个有 n 个记录的无序文件看成是由 n 个长度为 1 的有序子文件组成的文件，然后进行两两归并，如此重复，直至最后形成一个包含 n 个记录的有序文件为止。这种反复将两个有序文件归并成一个有序文件的排序方法称为两路归并排序。

3.6.1.7 基数排序

基数排序的思想是：设立 r 个队列，队列的编号分别为$0,1,2,\cdots,r-1$。首先按最低有效位的值，把 n 个关键字分配到这 r 个队列中；然后从小到大将各队列中关键字再依次收集起来；接着按次低有效位的值把刚收集起来的关键字再分配到 r 个队列中。重复上述收集过程，直至最高有效位，这样得到了一个从小到大有序的关键字序列。为了减少记录移动的次数，队列可以采用链式存储分配，称为链队列。每个链队列设有两个指针，分别指向队头和队尾。

对于 n 个记录，执行一次分配和收集的时间为$O(n+r)$，如果关键字有 d 位，则要执行 d 遍，所以总的运算时间为$O(d(n+r))$。基数排序适用于链式分配的记录的排序，是一种稳定的排序方法。

3.6.1.8 内部排序方法的比较和选择

1. 内部排序方法的比较

内部排序方法的比较参见表 3-1。

表 3-1 内部排序方法的比较

排序方法	最好时间	平均时间	最坏时间	辅助时间	稳定性
直接插入	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$	稳定
简单选择	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(1)$	不稳定
冒泡排序	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$	稳定
希尔排序	—	$O(n^{1.25})$	—	$O(1)$	不稳定
快速排序	$O(n\log_2 n)$	$O(n\log_2 n)$	$O(n^2)$	$O(n\log_2 n)$	不稳定

续表

排序方法	最好时间	平均时间	最坏时间	辅助时间	稳定性
堆排序	$O(n\log_2 n)$	$O(n\log_2 n)$	$O(n\log_2 n)$	$O(1)$	不稳定
归并排序	$O(n\log_2 n)$	$O(n\log_2 n)$	$O(n\log_2 n)$	$O(n)$	稳定
基数排序	$O(d(n+rd))$	$O(d(n+rd))$	$O(d(n+rd))$	$O(rd)$	稳定

2. 内部排序方法的选择

选择排序方法时需要考虑的因素有：①待排序的记录个数 n；②记录本身的大小；③关键字的分布情况；④对排序稳定性的要求；⑤语言工具的条件、辅助空间的大小。

依据这些因素，可以得到以下几点结论。

- 若待排序的记录数目 n 较小时，可采用插入排序和选择排序。
- 若待排序记录按关键字基本有序，则宜采用直接插入排序或冒泡排序。
- 当 n 很大且关键字的位数较少时，采用链式基数排序较好。
- 若 n 较大，则应采用时间复杂度为 $O(n\log_2 n)$ 的排序方法，如快速排序、堆排序或归并排序。

3.6.1.9　外部排序

常用的外部排序法是归并排序。这种方法一般分为两个阶段：在第一阶段，把文件中的记录分段读入内存，利用某种内部排序方法对这段记录进行排序并输出到外存的另一个文件中，在新文件中形成许多有序的记录段，称为归并段；在第二阶段，对第一阶段形成的归并段用某种归并方法进行一趟趟的归并，使文件的有序段逐渐加长，直到将整个文件归并为一个有序段时为止。

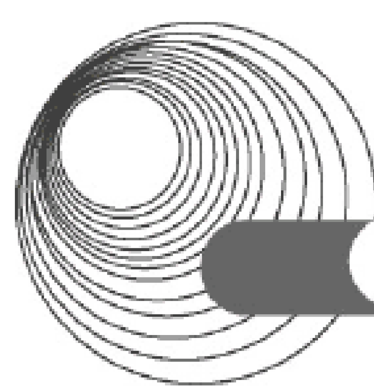
3.6.2　典型例题分析

例 1　给定 n 个整数构成的数组 $A=\{a_1, a_2, \cdots, a_n\}$ 和整数 x，判断 A 中是否存在两个元素 a_i 和 a_j，使得 $a_i+a_j=x$。为了求解问题，首先用归并排序算法对数组 A 进行从大到小排序；然后判断是否存在 $a_i+a_j=x$，具体的方法如下列伪代码所示，则求解该问题时排序算法应用了__(62)__算法设计策略，整个算法的时间复杂度为__(63)__。(2013 年上半年试题 62、63)

```
…
i=1;j=n
While i<j
  If ai+aj=x return true
   Else if ai+aj>x
        J--;
   Else
        I++;
  Return false;
```

(62) A. 分治　　B. 贪心　　C. 动态规划　　D. 回溯

(63) A. $O(n)$　　B. $O(n\lg n)$　　C. $O(n^2)$　　D. $O(n\lg^2 n)$



解析：分治算法的基本思想是将一个规模为 N 的问题分解为 K 个规模较小的子问题，这些子问题相互独立且与原问题性质相同。求出子问题的解，就可得到原问题的解。

答案：(62) A　(63) B

例 2　将数组{1,1,2,4,7,5}从小到大排序，若采用 (62) 排序算法，则元素之间需要进行的比较次数最少，共需要进行 (63) 次元素之间的比较。(2012 年下半年试题 62、63)

(62) A．直接　　B．归并　　C．堆　　D．快速

(63) A．5　　B．6　　C．7　　D．8

解析：直接插入排序算法的基本思想是将待排序数组分为两个部分，即已排好序部分和未排序部分。其主要步骤为：开始时，第一个元素在已排好序部分中，其余部分在未排序部分。然后依次从未排序部分中取出第一个元素，从后向前与排好序部分的元素进行比较并将其插入到已排好序部分的正确位置，直到所有元素排好序。当序列基本有序时，直接插入排序过程中元素比较的次数较少，当序列为逆序时，元素的比较次数最多。使用直接插入排序算法，数组{1,1,2,4,7,5}需要比较 6 次，依次为 1 与 1 比较、2 与 1 比较、4 与 2 比较、7 与 4 比较、5 与 7 比较、5 与 4 比较。

并归排序的基本思想是将待排序数组划分为子问题，对子问题求解，然后合并解。其主要步骤为：将数组分为两个相同规模的子数组，分别包含前 $n/2$ 个元素和后 $n/2$ 个元素；递归地排序这两个子数组；合并排好序的两个子数组，依次比较两个排好序的子数组的元素，得到整个数组的排好序的序列。使用直接插入排序算法，数组{1,1,2,4,7,5}需要比较 8 次。

答案：(62) A　(63) B

例 3　递增序列 $A(a_1, a_2, \cdots, a_n)$和 $B(b_1, b_2, \cdots, b_n)$的元素互不相同，若需将它们合并为一个长度为 $2n$ 的递增序列，则当最终的排列结果为 (61) 时，归并过程中元素的比较次数最多。(2012 年上半年试题 61)

(61) A. $a_1, a_2, \cdots, a_n, b_1, b_2, \cdots, b_n$

B. $b_1, b_2, \cdots, b_n, a_1, a_2, \cdots, a_n$

C. $a_1, b_1, a_2, b_2, \cdots, a_i, b_i, \cdots, a_n, b_n$

D. $a_1, a_2, \cdots, a_i/2, b_1, b_2, \cdots, b_i/2, a_i/2+1, a_i/2+2, \cdots a_n, b_i/2+1, b_i/2+2, \cdots, b_n$

解析：归并排序是将两个排好序的序列合并成一个有序的序列。由选项 A 给出的结果可知，递增序列 B 的每一个元素都比 A 中的元素要大，也就是说 $a_i(1 \leqslant i \leqslant n)$比 b_1 小，在排序的过程中，只需要将 a_i 与 b_1 进行比较，共比较了 n 次。由选项 B 给出的结果可知，递增序列 B 的每一个元素都比 A 中的元素要小，在排序的过程中，只需要将 $b_i(1 \leqslant i \leqslant n)$与 a_1 进行比较，共比较了 n 次。由选项 C 给出的结果可知，$a_i<b_i<a_{i+1}$，在排序的过程中，将 a_1 与 b_1 进行比较，a_1 小，然后将 a_2 与 b_1 进行比较，a_2 大，则已排好的部分为 a_1b_1，共比较了两次；然后将 a_2 与 b_2 进行比较，a_2 小，再将 a_3 与 b_2 进行比较，a_3 大，则已排好的部分为 $a_1b_1a_2b_2$，共比较了 4 次；以此类推，完全排好时共比较了 $2(n-1)+1=2n-1$ 次。由选项 A 给出的结果可知，递增序列 B 的前 $i/2$ 个元素都比 A 中的前 $i/2$ 个元素要大，但比 A 中的后 $i/2$ 个元素要小，B 的后 $i/2$ 个元素都比 A 中的后 $i/2$ 个，因此在排序的时候，a 的前 $i/2$ 个元素只需与 b_1 进行比较，当比较到 $a_{i/2+1}$ 时，$a_{i/2+1}$ 比 b_1 大，则已排好的部分为 $a_1, a_2, \ldots, a_{i/2}$，共比较了 $i/2+1$ 次；然后将 $b_2, b_3, \ldots, b_{i/2}$ 与 $a_{i/2+1}$ 进行比较，都比 $a_{i/2+1}$ 小，当比较到 $b_{i/2+1}$ 时，

$b_{i/2+1}$ 比 $a_{i/2+1}$ 大，则已排好的部分为 $a_1, a_2, \cdots, a_{i/2}, b_1, b_2, \cdots, b_{i/2}$，共比较了 i+1 次；然后将 $a_{i/2+2}, \cdots, a_n$ 分别与 $b_{i/2+1}$ 进行比较，共比较了 $n-i/2-2$ 次，完全排好时共比较了 $i+1+n-i/2-2=n+i/2-1$ 次。

答案： C

例 4　在某应用中，需要先排序一组大规模的记录，其关键字为整数。若这组记录的关键字基本上有序，则适宜采用 (64) 排序算法。若这组记录的关键字的取值均在 0～9 之间(含)，则适宜采用 (65) 排序算法。(2015 年上半年试题 64、65)

(64) A．插入　　B．归并　　C．快速　　D．计数

(65) A．插入　　B．归并　　C．快速　　D．计数

解析： 插入排序的基本方法是：每步将一个待排序的记录按其关键字的大小插到前面已经排序的序列中的适当位置，直到全部记录插入完毕为止。关键字基本有序时，宜采用插入排序算法，每趟排序移动元素较少。

计数排序是一种算法复杂度为 $O(n)$的排序方法，适合于小范围集合的排序。计数排序是一个类似于桶排序的排序算法，其优势是对已知数量范围的数组进行排序。它创建一个长度为这个数据范围的数组 C，C 中每个元素记录要排序数组中对应记录的出现个数。这个算法于 1954 年由 Harold H. Seward 提出。

答案： (64) A　　(65) D

例 5　在 n 个数的数组中确定其第 $i(1 \leqslant i \leqslant n)$小的数时，可以采用快速排序算法中的划分思想，对 n 个元素划分，先确定第 k 小的数，根据 i 和 k 的大小关系进一步处理，最终得到第 i 小的数。划分过程中，最佳的基准元素选择的方法是选择待划分数组的 (64) 元素。此时，算法在最坏情况下的时间复杂度为(不考虑所有元素均相等的情况) (65) 。(2015 年上半年试题 64、65)

(64) A. 第一个　　B. 最后一个　　C. 中位数　　D. 随机一个

(65) A. $\Theta(n)$　　B. $\Theta(\lg n)$　　C. $\Theta(n\lg n)$　　D. $\Theta(n^2)$

解析： 快速排序的 3 个步骤如下。

(1) 选择基准。在待排序列中，按照某种方式挑出一个元素，作为“基准”(Pivot)。

(2) 分割操作。以该基准在序列中的实际位置，把序列分成两个子序列。此时，在基准左边的元素都比该基准小，在基准右边的元素都比该基准大。

(3) 递归地对两个序列进行快速排序，直到序列为空或者只有一个元素。

对于分治算法，当每次划分时，算法若都能分成两个等长的子序列时，那么分治算法效率会达到最大。也就是说，基准的选择是很重要的。选择基准的方式决定了分割后两个子序列的长度，进而对整个算法的效率产生决定性影响。最佳的划分是将待排序的序列分成等长的子序列，最佳的状态可以使用序列的中间值，也就是第 $N/2$ 个数。可是，这很难算出来，并且会明显减慢快速排序的速度。这样的中值估计可以通过随机选取 3 个元素并用它们的中值作为枢纽元素而得到。

排序算法的平均复杂度为 $\Theta(n\lg n)$，最坏情况下为 $\Theta(n^2)$。

答案： (64) C　　(65) D

例 6　用某排序方法对一元素序列进行非递减排序时，若该方法可保证在排序前后排序码相同者的相对位置不变，则称该排序方法是稳定的。简单选择排序法的排序方法是不稳

定的，__(61)__可以说明这个性质。(2015 年上半年试题 61)

(61) A. 21　48　21*　63　17

B. 17　21　21*　48　63

C. 63　21　48　21*　17

D. 21*　17　48　63　21

解析： 简单选择排序的基本方法是：①从待排序序列中，找到关键字最小的元素；②如果最小元素不是待排序序列的第一个元素，将其和第一个元素互换；③从余下的 N−1 个元素中，找出关键字最小的元素，重复①②步，直到排序结束。

对于选项 B，21 和后面 4 个元素相比，17 是比 21 小，与 17 交换位置；待排序序列为 48、21*、63、21，48 与 21*相比，21*小，48 与其互换位置，21*再与后面的元素进行比较，由于 21*不比后面的元素小，不需要再换位置；待排序序列为 48、63、21，排序后为 21、48、63。最终的排序结果为 17、21*、21、48、63。排序前 21 在 21*之前，排序后 21 在 21*之后，可见排序方法是不稳定的。

答案： A

3.6.3 同步练习

1. 对 n 个元素的数组进行_____，其平均时间复杂度和最坏情况下的时间复杂度都是 $O(n\log_2 n)$。

A．希尔排序　B．快速排序　C．堆排序　D．选择排序

2. 对于具有 n 个元素的一个数据序列，若只需得到其中第 k 个元素之前的部分排序，最好采用__(1)__，使用分治(Divide and Conquer)策略的是__(2)__算法。

(1) A．希尔排序　B．直接插入排序

C．快速排序　D．堆排序

(2) A．冒泡排序　B．插入排序

C．快速排序　D．堆排序

3. _____在其最好情况下的算法时间复杂度为 $O(n)$。

A．插入排序　B．归并排序　C．快速排序　D．堆排序

4. 快速排序算法采用的设计方法是_____。

A．动态规划法　B．分治法　C．回溯法　D．分支定界法

5. _____在其最好情况下的算法时间复杂度为 $O(n)$ 。

A．插入排序　B．归并排序　C．快速排序　D．堆排序

6. 若排序前后关键字相同的两个元素相对位置不变，则称该排序方法是稳定的。_____排序是稳定的。

A．归并　B．快速　C．希尔　D．堆

7. 利用逐点插入建立序列(50, 72, 43, 85, 75, 20, 35, 45, 65, 30)对应的二叉排序树以后，查找元素 30 要进行_____次元素间的比较。

A．4　B．5　C．6　D．7

8. 在最好和最坏情况下的时间复杂度均为 $O(n\log_2 n)$且稳定的排序方法是_____。

A．基数排序　　B．快速排序　　C．堆排序　　D．归并排序

9. 以比较为基础的排序算法在最坏情况下的计算时间下界为_____。

A．$O(n)$　　B．$O(n^2)$　　C．$O(\log_2 n)$　　D．$O(n\log_2 n)$

10. 若对 27 个元素只进行 3 趟多路归并排序，则选取的归并路数为_____。

A．2　　B．3　　C．4　　D．5

11. 堆是一种数据结构，_____是堆。

A．(10, 50, 80, 30, 60, 20, 15, 18)　　B．(10, 18, 15, 20, 50, 80, 30, 60)

C．(10, 15, 18, 50, 80, 30, 60, 20)　　D．(10, 30, 60, 20, 15, 18, 50, 80)

3.6.4 同步练习参考答案

1．C　　2．(1) D　(2) C　　3．A　　4．B

5．A　　6．A　　7．B　　8．D

9．D　　10．C　　11．B

3.7 本 章 小 结

本章知识点在 2013 年的新大纲中改动不大，只是有一些描述方面的调整。

本章主要要求考生掌握数据结构的基础知识，包括数组、线性表、链表、队列、栈、树、图等的定义、存储和操作；哈希函数；查找算法和排序算法。

数据结构是软考的一个考查重点，每次考试除了下午部分的设计题目外，上午试题中大概考 7～8 道小题。

本章知识点较多，考查的重点是树、二叉树、图和排序算法。其中特别是排序算法的题比较多，复习时应注意重点掌握。

3.8 达标训练题及参考答案

3.8.1 达标训练题

1. 从下列有关树的叙述中，选出 5 条叙述正确的_____。

A．一棵二叉树的层次遍历方法只有前序法和后序法两种

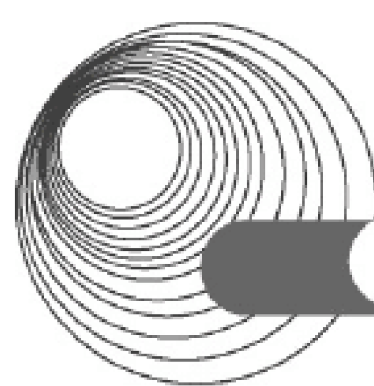
B．在哈夫曼树中，外部节点的个数比内部节点个数多 1

C．完全二叉树一定是平衡二叉树

D．在二叉树的前序序列中，若节点 u 在节点 v 之前，则 u 一定是 v 的祖先

E．在查找树中插入一个新节点，总是插入到叶节点下面

F．树的后序序列和其对应的二叉树的后序序列的结果是一样的



G. 对 B-树删除某一关键字值时，可能会引起节点的分裂

H. 在含有 *n* 个节点的树中，边数只能是 *n*−1 条

I. 最佳查找树就是检索效率最高的查找树

J. 中序遍历二叉链存储的二叉树时，一般要用堆栈，中序遍历检索二叉树时，也必须使用堆栈

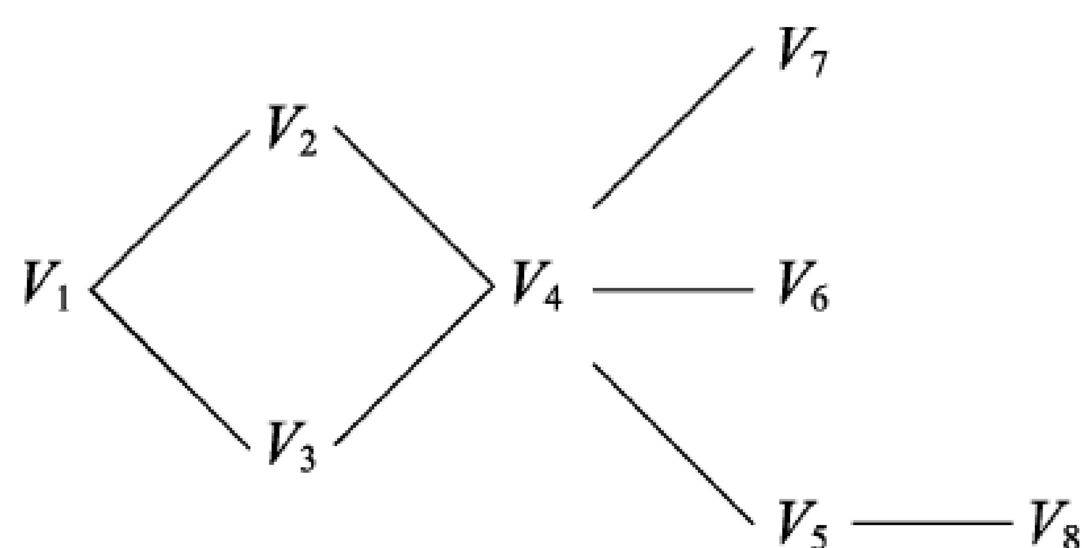
2. 给定数据结构(V, E)，V 为节点的有限集合，$V=\{V_1,V_2,V_3,V_4,V_5,V_6,V_7,V_8\}$，$E$ 是 V 上关系的集合。

$E=\{<V_1,V_2>, <V_3,V_4>, <V_5,V_8>, <V_5,V_6>, <V_1,V_3>, <V_4,V_7>, <V_4,V_5>, <V_2,V_4>, <V_4,V_6>\}$ 所对应的图形是(1)，这是(2)。

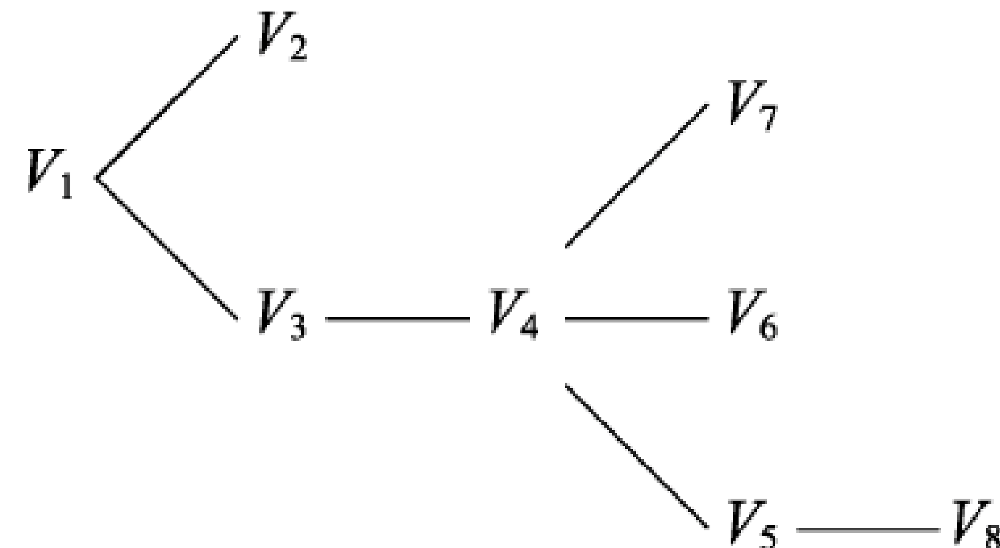
图的存储结构主要有邻接表和(3)，若用邻接表来存储一个图，则需要保存一个(4)存储的节点表和若干个(5)上存储的关系表(又称边表)。

(1)

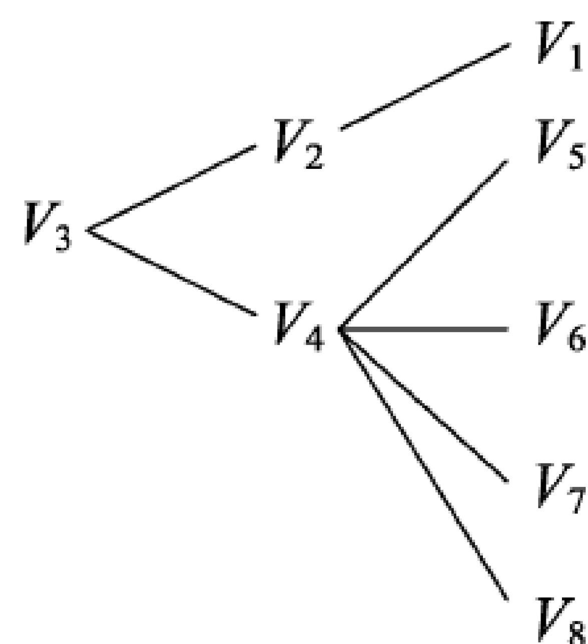
A.



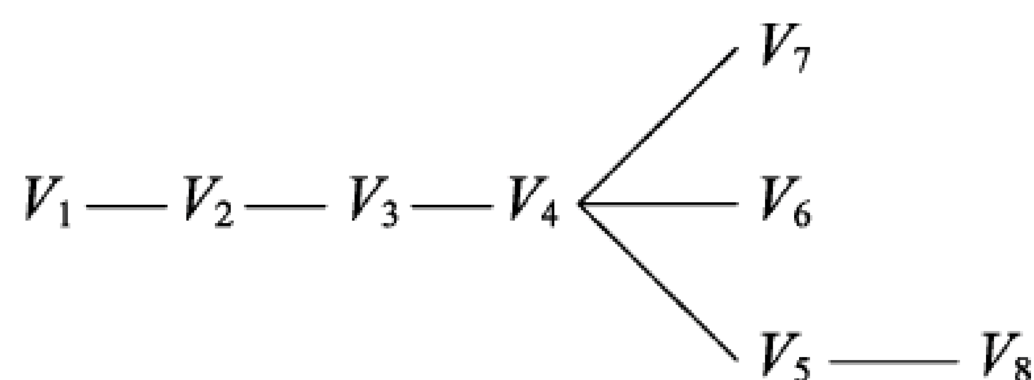
B.



C.



D.



(2) A. 树　B. 无向图　C. 有向图　D. 无向图

(3) A. 转移矩阵　B. 邻接矩阵　C. 状态矩阵　D. 优先矩阵

(4)～(5) A. 顺序　B. 链接　C. 散列　D. 分块

3. 对于给定的一组关键字(12,2,16,30,8,28,4,10,20,6,18)，按照下列算法进行递增排序，写出每种算法第一趟排序后得到的结果：希尔排序(增量为 5)得到 (1) ，快速排序(选第一个记录为基准元素)得到 (2) ，链式基数(基数为 10)排序得到 (3) ，二路归并排序得到 (4) ，堆排序得到 (5) 。

(1) A. 2, 4, 6, 8, 10, 12, 16, 18, 20, 28, 30

B. 6, 2, 10, 4, 8, 12, 28, 30, 20, 16, 18

C. 12, 2, 10, 20, 6, 18, 4, 16, 30, 8, 28

D. 30, 10, 20, 12, 2, 4, 16, 6, 8, 28, 18

(2) A. 10, 6, 18, 8, 4, 2, 12, 20, 16, 30, 28

B. 6, 2, 10, 4, 8, 12, 28, 30, 20, 16, 10

C. 2, 4, 6, 8, 10, 12, 16, 18, 20, 28, 30

D．6, 10, 8, 28, 20, 18, 2, 4, 12, 30, 16

(3)　A．10, 6, 18, 8, 4, 2, 12, 20, 16, 30, 28

B．1, 12, 10, 20, 6, 18, 4, 16, 30, 8, 28

C．2, 4, 6, 8, 10, 12, 16, 18, 20, 28, 30

D．30, 10, 20, 12, 2, 4, 16, 6, 8, 28, 18

(4)　A．2, 12, 16, 8, 28, 30, 4, 6, 10, 18, 20

B．2, 12, 16, 30, 8, 28, 4, 10, 6, 20, 18

C．12, 2, 16, 8, 28, 30, 4, 6, 10, 28, 18

D．12, 2, 10, 20, 6, 18, 4, 16, 30, 8, 28

(5)　A．30, 28, 20, 12, 18, 16, 4, 10, 2, 6, 8

B．20, 30, 28, 12, 18, 4, 16, 10, 2, 8, 6

C．2, 6, 4, 10, 8, 28, 16, 30, 20, 12, 18

D．2, 4, 10, 6, 12, 28, 16, 20, 8, 30, 18

4.　在内部排序中，通常要对被排序数据序列进行多趟扫描。各种排序方法有其不同排序实施过程和(时间)复杂性。

对给定的整数序列(541, 132, 984, 746, 518, 181, 946, 314, 205, 827)进行从小到大的排序时，采用冒泡排序和直接选择排序时若先选出大元素，则第一趟扫描结果分别是__(1)__和__(2)__；采用快速排序(以中间元素 518 为基准)的第一趟扫描结果是__(3)__。

设被排序数据序列有 N 个元素，冒泡排序和直接选择排序的复杂性是__(4)__；快速排序的复杂性是__(5)__。

(1)～(3)　A．(181, 132, 314, 205, 541, 518, 946, 827, 746, 984)

B．(541, 132, 827, 746, 518, 181, 946, 314, 205, 984)

C．(205, 132, 314, 181, 518, 746, 946, 984, 541, 827)

D．(541, 132, 984, 746, 827, 181, 946, 314, 205, 518)

E．(132, 541, 746, 518, 181, 946, 314, 205, 827, 984)

F．(132, 541, 746, 984, 181, 518, 314, 946, 205, 827)

(4)～(5)　A．$O(n\log_2 n)$　B．$O(n)$　C．$O(\log_2 n)$

D．$O(n^2)$　E．$O((\log_2 n)^2)$　F．$O(n^2\log_2 n)$

5.　已知某二叉树的中序、层序序列分别为 DBAFCE、FDEBCA，则该二叉树的后序序列为______。

A．BCDEAF　B．ABDCEF　C．DBACEF　D．DABECF

6.　在二叉树的顺序存储中，每个节点的存储位置与其父节点、左右子树节点的位置都存在一个简单的映射关系，因此可与三叉链表对应。若某二叉树共有 n 个节点，采用三叉链表存储时，每个节点的数据域需要 dB，每个指针域占用 4B，若采用顺序存储，则最后一个节点下标为 k(起始下标为 1)，那么______时采用顺序存储更节省空间。

A．$d<12n/(k-n)$　B．$d>12n/(k-n)$

C．$d<12n/(k+n)$　D．$d>12n/(k+n)$

7.　由元素序列(27, 16, 75, 38, 51)构造平衡二叉树，则首次出现的最小不平衡子树的根(即离插入节点最近且平衡因子的绝对值为 2 的节点)为______。

A．27　　B．8　　C．51　　D．75

3.8.2　参考答案

1．B　C　E　H　I

2．(1) A　(2) C　(3) B　(4) A　(5) B

3．(1) C　(2) B　(3) D　(4) B　(5) C

4．(1) E　(2) B　(3) C　(4) D　(5) A

5．B　6．A　7．D

第 4 章　操作系统知识

大纲要求：

- 操作系统的内核(中断控制)、进程、线程概念。
- 处理机管理(状态转换、共享与互斥、分时轮转、抢占、死锁)。
- 存储管理(主存保护、动态链接分配、分段、分页、虚存)。
- 设备管理(I/O 控制、假脱机)。
- 文件管理(文件目录、文件组织、存取方法、存取控制、恢复处理)。
- 作业管理(作业调度、作业控制语言(JCL)、多道程序设计)。
- 汉字处理、多媒体处理、人机界面。
- 网络操作系统和嵌入式操作系统的基础知识。
- 操作系统的配置。

4.1　操作系统基础知识

4.1.1　考点辅导

4.1.1.1　操作系统的定义和作用

1. 操作系统的定义

操作系统(Operating System，OS)是计算机系统中的一个系统软件，它管理和控制计算机系统的硬件和软件资源，合理地组织计算机的工作流程，控制程序的执行，并且向用户提供一个良好的工作环境和友好的接口。

2. 操作系统的作用

操作系统具有以下作用。

(1) 通过资源管理，提高计算机系统的效率。

(2) 改善人机界面，向用户提供友好的工作环境。

4.1.1.2　操作系统的特征与功能

1. 操作系统的特征

操作系统主要有并发性、共享性、虚拟性和不确定性这 4 个基本特征。

(1) 并发性。对计算机系统而言，并发指的是宏观上系统在一段时间内有多个程序在同时运行，微观上看是串行运行的。

(2) 共享性。共享是指系统中的资源可供多个并发执行的程序共同使用。

(3) 虚拟性。虚拟是指使用分时技术，将一个物理上的设备虚拟成多个逻辑上的设备。

(4) 不确定性。由于程序的并发运行而导致系统内的各种进程错综复杂，与这些进程有关的事件发生的时间都不可预测，如果处理不当，将导致系统出错。

2. 操作系统的功能

操作系统具有以下功能。

1) 处理机管理

处理机管理实际上是指对处理机执行“时间”的管理，采用多道程序等技术将 CPU 真正合理地分配给每个任务。常用的资源管理单位有进程和线程。

(1) 进程控制：创建、撤销、挂起、改变运行优先级等。

(2) 进程同步：协调并发进程之间的推进步骤，以协调资源共享。

(3) 进程通信：在进程之间传送数据，以协调进程间的协作。

(4) 进程调度：作业和进程的运行切换，以充分利用处理机资源和提高系统性能。

2) 文件管理

文件管理(信息管理)包括以下内容。

(1) 文件存储空间管理：解决如何存放信息，以提高空间利用率和读写性能。

(2) 目录管理：解决信息检索问题。

(3) 文件的读写管理和存取控制：通过系统设置用户口令、对用户分类、通过设置文件权限解决信息安全问题。

(4) 软件管理：软件的版本、相互依赖关系、安装和卸载等。

3) 存储管理

存储管理主要是对主存空间进行管理。

(1) 存储分配与回收：操作系统为每个程序分配存储空间，当程序运行完毕时回收存储空间。

(2) 存储保护：保证进程之间互不干扰、相互保密。

(3) 地址映射：进行逻辑地址到主存物理地址的映射。

(4) 主存扩充：提高主存利用率，扩大进程的主存空间。

4) 设备管理

设备管理的目标是方便设备使用，提高 CPU 与 I/O 设备的利用率。

(1) 设备操作：利用设备驱动程序完成对设备的操作。

(2) 设备独立性：提供统一的 I/O 设备接口，使应用程序独立于物理设备，提高可适应性；在同样的接口和操作下完成不同的内容。

(3) 设备分配与回收：在多用户间共享 I/O 设备资源。

(4) 缓冲区管理：匹配 CPU 和外设的速度，提高两者的利用率。

5) 作业管理

作业管理包括任务、界面管理、人机交互、图形界面、语音控制和虚拟现实等。操作系统提供系统命令一级的接口，供用户用于组织和控制自己的作业运行。操作系统还提供编程级的接口，供用户程序和系统程序调用操作系统功能。

4.1.1.3 操作系统的类型

1. 批处理操作系统

批处理操作系统分为单道批处理和多道批处理两种。

(1) 单道批处理操作系统。单道批处理操作系统是一种早期的操作系统，该系统可以提交多个作业。“单道”的含义是指一次只有一个作业装入内存执行。作业由用户程序、数据和作业说明书(作业控制语言)三部分组成。当一个作业运行结束后，随即自动调入同批的下一个作业运行，从而节省了作业之间的人工干预时间，提高了资源的利用率。

(2) 多道批处理操作系统。多道批处理操作系统允许多个作业装入内存执行，在任意一个时刻，作业都处于开始点和终止点之间。每当运行中的一个作业因输入输出操作需要调用外部设备时，就把 CPU 及时交给另一道等待运行的作业，从而将主机与外部设备的工作由串行改变为并行，进一步避免了因主机等待外设完成任务而白白浪费宝贵的 CPU 时间。多道批处理系统主要有 3 个特点，即多道、宏观上并行运行、微观上串行运行。

2. 分时操作系统

分时操作系统是将 CPU 的工作时间划分为许多很短的时间片，轮流为各个终端用户服务。分时操作系统的特点如下。

(1) 多路性。允许在一台主机上同时连接多台联机终端，系统按分时原则为每个用户服务。

(2) 独立性。每个用户各占一个终端，批次独立操作，互不干扰。因此，用户感觉就像自己独占了主机。

(3) 交互性。用户可通过终端与系统进行人机对话。

(4) 及时性。用户的请求在很短的时间内获得响应。

UNIX 系统是典型多用户、多任务的分时操作系统。

3. 实时操作系统

实时操作系统可分为以下两类。

(1) 实时控制系统。主要用于生产过程的自动控制，实验数据的自动采集、武器的控制，包括火炮自动控制、飞机自动驾驶、导弹的制导系统。

(2) 实时信息处理系统。主要用于实时信息处理，如飞机订票系统、情报检索系统。

实时系统的主要特点如下。

① 快速的响应时间。实时系统是为了提高系统响应时间而设计的操作系统，特别是实时控制系统，对外部时间的响应要十分及时。

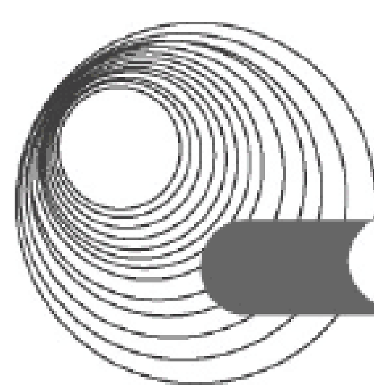
② 有限的交互能力。实时系统一般是专用系统，它能提供人机交互方式，但用户只能访问系统中某些特定的专用服务程序，不能像分时系统那样为终端用户提供多方面服务。

③ 高可靠性。实时系统要求高度可靠，因此实时系统中往往都采用双机系统和多级容错措施来保证系统和数据的安全。

实时系统与分时系统的主要区别：实时系统与分时系统除了应用环境不同之外，在系统设计目标、交互性强弱、响应时间的敏感程度等方面也不同。

4. 网络操作系统

网络操作系统是使联网的计算机能方便而有效地共享网络资源，为网络用户提供所需各种服务的软件和有关协议的集合。其功能主要包括：高效、可靠的网络通信；对网络中的共享资源进行有效管理；提供电子邮件、文件传输、共享硬盘、打印机等服务；网络安全管理；提供互操作能力。



5. 分布式操作系统

分布式操作系统是网络操作系统的更高级形式，它可以保持网络操作系统所拥有的全部功能，同时又具有透明性、可靠性和高性能。网络操作系统与分布式操作系统最大的差别是：网络操作系统的用户必须知道网址，而分布式系统用户则不必知道计算机的确切地址；分布式操作系统负责整个系统的资源分配，通常能很好地隐藏系统内部的实现细节，如对象的物理位置、并发控制等，这些对用户都是透明的。

6. 微机操作系统

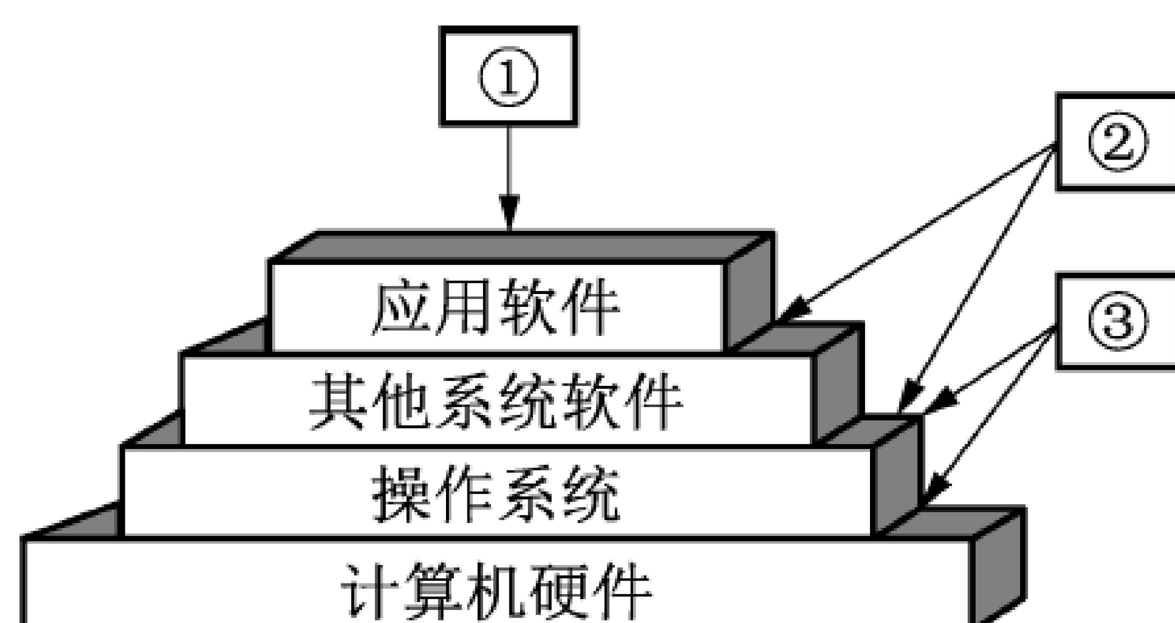
微机操作系统是指配置在微型计算机上的操作系统。常用的微机操作系统有 DOS、Windows、OS/2、SCO UNIX 和 Linux 等(注：UNIX 也可写成 Unix)。其中，Microsoft 公司开发的单用户单任务操作系统 MS-DOS 是首先在 IBM-PC 上使用的微机操作系统。MS-DOS 操作系统是事实上的 16 位微机单用户单任务操作系统的标准。多任务操作系统 Windows 98/NT/2000/XP 是 Microsoft 公司开发的一系列图形用户界面的多任务、多线程的操作系统。SCO UNIX 是 SCO 公司将运行于大、中、小型机上的 UNIX 操作系统移植到微机上的多用户多任务操作系统。Linux 操作系统是一个遵循标准操作系统界面的免费操作系统。

7. 嵌入式操作系统

嵌入式操作系统运行在嵌入式智能芯片环境中，对整个智能芯片及其控制的各种部件和装置等资源进行统一协调、处理、指挥和控制。

4.1.2 典型例题分析

例 操作系统是裸机上的第一层软件，其他系统软件(如(23)等)和应用软件都是建立在操作系统基础上的。如下图所示的①、②、③分别表示(24)。(2009 年下半年试题 23、24)



(23) A．编译程序、财务软件和数据库管理系统软件
　　B．汇编程序、编译程序和 Java 解释器
　　C．编译程序、数据库管理系统软件和汽车防盗程序
　　D．语言处理程序、办公管理软件和气象预报软件

(24) A．应用软件开发者、最终用户和系统软件开发者
　　B．应用软件开发者、系统软件开发者和最终用户
　　C．最终用户、系统软件开发者和应用软件开发者
　　D．最终用户、应用软件开发者和系统软件开发者

解析：本题主要考查系统软件和应用软件的区别。

应用软件是专门为某一应用目的而编制的软件，一般包括文字处理软件(如 Word、WPS

等)、信息处理软件(用于输入、存储、修改、检索各种信息，如工资管理软件、人事管理软件、仓库管理软件、计划管理软件等)、辅助设计软件(用于高效地绘制、修改工程图纸，进行设计中的常规计算，帮助人寻求设计方案)、实时控制软件(用于随时搜集生产装置、飞行器等的运行状态信息，以此为依据按预定的方案实施自动或半自动控制，安全、准确地完成任务)。

系统软件主要负责数据是如何输入输出的以及对硬件的管理等，主要有以下几种。

(1) 操作系统：DOS、Windows、UNIX 等。

(2) 数据库管理系统：有组织地、动态地存储大量数据，使人们能方便、高效地使用这些数据。现在比较流行的数据库有 FoxPro、DB-2、Access、SQL Server 等。

(3) 编译软件：源程序与机器代码的转换，完成这种翻译的软件称为高级语言编译软件，通常将其归入系统软件。目前常用的高级语言有 Visual Basic、C++、Java 等，它们各有特点，分别适用于编写某一类型的程序，它们都有各自的编译软件。

应用软件是为最终用户服务的，因此①应为最终用户，②是工作在其他系统软件和操作系统基础上，应该为应用软件开发者，而③是工作在操作系统和计算机硬件上，应为系统软件开发者。

答案：(23) B　　(24) D

4.1.3　同步练习

下面关于操作系统的叙述中，正确的是______。

A．批处理作业必须具有作业控制信息

B．分时系统不一定都具有人-机交互功能

C．从响应时间的角度看，实时系统与分时系统差不多

D．由于采用了分时技术，用户可以独占计算机的资源

4.1.4　同步练习参考答案

A

4.2　进 程 管 理

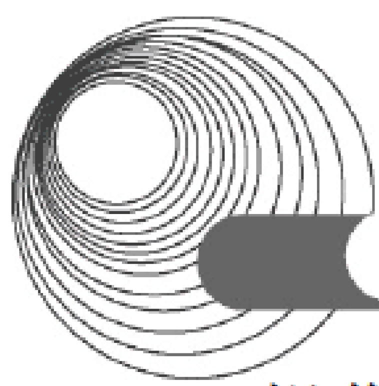
4.2.1　考点辅导

4.2.1.1　基本概念

1．程序与进程

1)　程序

前趋图是一个有向无循环图，图由节点和节点间的有向边组成，节点代表各程序段的



操作，而节点间的有向边表示两程序段操作之间存在的前趋关系(“→”)。两程序段 P_i 和 P_j 的前趋关系表示成 $P_i \rightarrow P_j$，其中 P_i 是 P_j 的前趋，P_j 是 P_i 的后继，其含义是 P_i 执行完毕才能由 P_j 执行。

(1) 程序顺序执行时的特征。

- 顺序性。程序中的各程序段严格按照规定的顺序执行。
- 封闭性。指程序运行时系统内各资源只受该程序控制，执行结果不受外界因素影响。
- 可再现性。只要程序执行环境和初始条件相同，运行结果就相同。

(2) 程序并发执行的特征。

- 失去了程序的封闭性。
- 程序和机器执行程序的活动不再一一对应。
- 并发程序间无相互制约性。

2) 进程

进程通常是由程序、数据及进程控制块(PCB)组成的。进程的程序部分描述了进程需要完成的功能，进程数据集合部分包括程序执行时所需的数据及工作区。

进程控制块是进程的描述信息和控制信息，是进程动态特性的集中反映，也是进程存在的唯一标志。进程控制块包含的主要内容有进程标志符、状态、位置信息、控制信息、队列指针、优先级、现场保护区及其他。PCB 是操作系统中最主要的数据结构之一，既是进程存在的标志和调度的依据，又是进程可以被打断并能恢复运行的基础。操作系统通过 PCB 管理进程，一般 PCB 是常驻主存的，尤其是调度信息必须常驻主存。

2. 进程的状态及其转换

1) 三态模型

三态模型中最基本的状态有 3 种，即运行、就绪和阻塞。

- 运行。进程正在处理机上运行。对于单处理机系统，处于运行状态的进程只有一个。
- 就绪。进程具备运行条件，但尚未运行。
- 阻塞。进程因发生某事件而暂停执行时的状态。

在进程运行过程中，由于自身进展情况及外界环境的变化，这 3 种基本状态可以在一定的条件下相互转换。进程的状态及转换如图 4-1 所示。

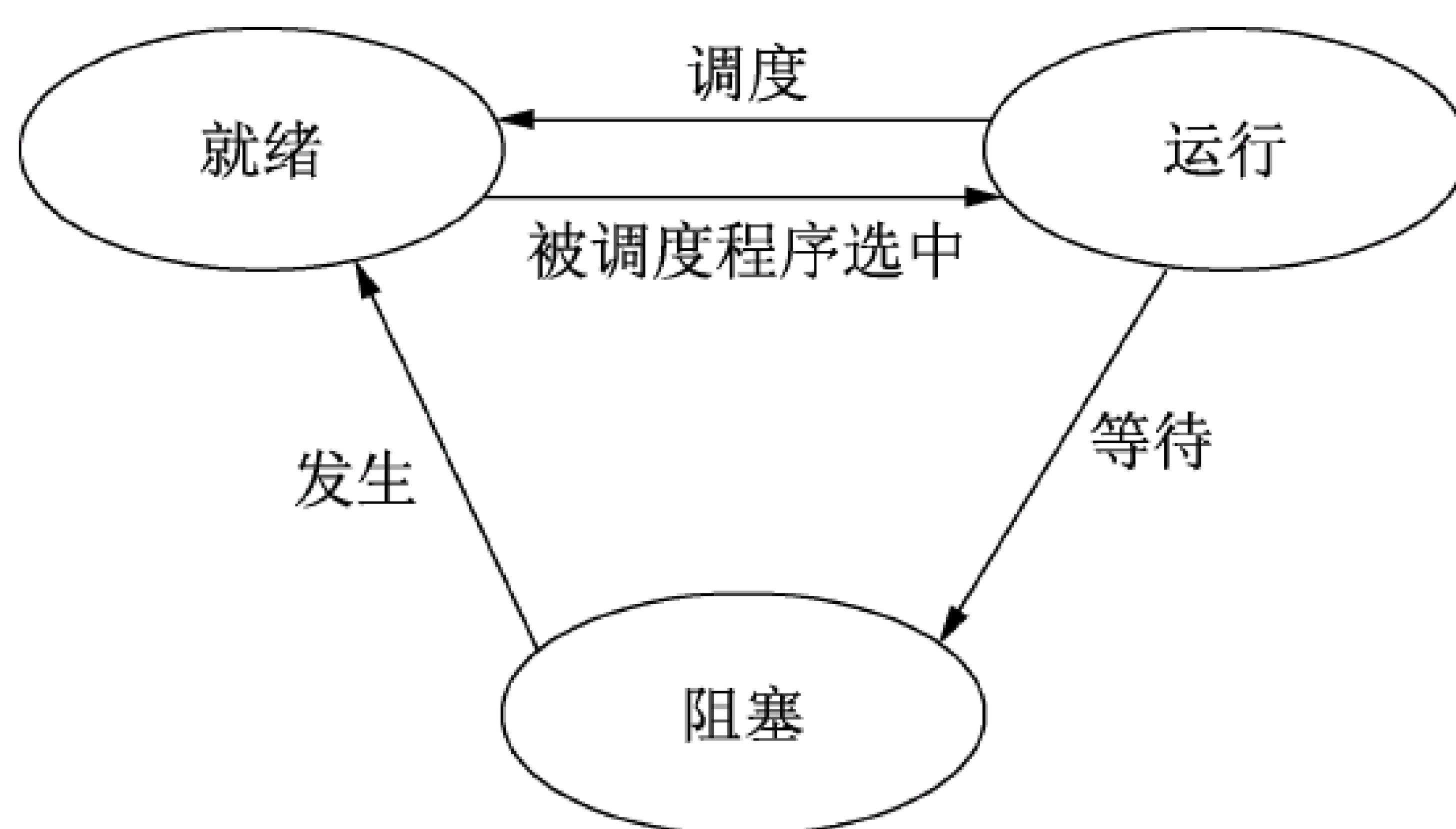


图 4-1 进程的状态及其转换

2) 五态模型

五态模型在三态模型的基础上增加了新建态和终止态。新建态是一个进程刚刚被创建

还没有被提交，并等待系统完成创建进程的所有必要信息状态。终止态是指当一个进程已经正常结束或异常结束，操作系统进行善后处理并且释放主存的状态。

3)　具有挂起状态的进程状态

由于进程的不断创建，系统资源特别是主存资源已不能满足所有进程的运行要求，这时就必须将某些进程挂起，放到磁盘对换区，暂时不参加调度，以平衡系统负载。具有挂起状态的进程状态包括活跃就绪、静止就绪、活跃阻塞、静止阻塞。

4.2.1.2　进程的控制

进程控制就是对系统中所有进程从创建到消亡的全过程实施有效的控制。为此，操作系统设置了一套控制机构，该机构的主要功能包括创建一个新进程，撤销一个已经运行完的进程，改变进程的状态，实现进程间的通信。进程控制是由操作系统内核中的原语实现的。内核是计算机系统硬件的首次延伸，是基于硬件的第一层软件扩充，它为系统对进程进行控制和管理提供了良好的环境。

原语是指由若干条机器指令组成的，用于完成特定功能的程序段。原语的特点是在执行时不能被分割，即原子操作——要么都做，要么都不做。内核中所包含的原语主要有进程控制原语、进程通信原语、资源管理原语以及其他方面的原语。属于进程控制方面的原语有进程创建原语、进程撤销原语、进程挂起原语、进程激活原语、进程阻塞原语以及进程唤醒原语等。不同的操作系统内核所包含的功能不同，但大多数操作系统的内核都包含支撑功能和资源管理的功能。

4.2.1.3　进程间的通信

1. 同步与互斥

同步是合作进程间的直接制约问题，互斥是申请临界资源进程间的间接制约问题。

1)　同步

相互合作的进程需要在某些确定点上协调它们的工作，当一个进程到达这些点后，除非另一个进程已经完成某些操作；否则就不得不停下来等待这些操作结束。这就是进程间的同步。

2)　互斥

在多道程序系统中，各进程可以共享各类资源，但有些资源一次只能供一个进程使用，称为临界资源。这就产生了进程间的间接制约问题——互斥。

3)　临界区管理的原则

临界区是进程中对临界资源实施操作的那段程序。互斥临界区管理的原则是：有空即进，无空则登；有限等待，让权等待。

2. 信号量与 P、V 操作

信号量机制主要有整型信号量、记录性信号量、信号量集机制。

1)　整型信号量与 P、V 操作

信号量是一个整型变量，根据控制对象的不同赋予不同的值。信号量可分为以下两类。

(1)　公用信号量。实现进程间的互斥，初值为 1 或资源的数目。

(2)　私用信号量。实现进程间的同步，初值为 0 或某个正整数。

信号量 S 的物理意义为：$S \geq 0$，表示某资源的可用数；$S<0$，其绝对值表示阻塞队列中等待该资源的进程数。P、V 操作是实现进程同步与互斥的常用方法。

P 操作定义：$S:=S-1$，若 $S \geq 0$，则执行 P 操作的进程继续执行；否则，若 $S<0$，则置该进程为阻塞状态，并将其插入阻塞队列。

V 操作定义：$S:=S+1$，若 $S>0$，则执行 V 操作的进程继续执行；否则，若 $S \leq 0$，则从阻塞状态唤醒一个进程，并将其插入就绪队列，执行 V 操作的进程继续执行。

2) 利用 P、V 操作实现进程的互斥

令信号量的初值为 1，当进程进入临界区时执行 P 操作，退出临界区时执行 V 操作。则进入临界区的代码段如下。

```
P(mutex);
临界区
V(mutex);
```

3) 利用 P、V 操作实现进程的同步

进程的同步是由于进程间合作而引起的相互制约问题。要实现进程的同步，可用一个信号量与消息联系起来。当信号量的值为 0 时表示消息未产生，当信号量的值为非 0 时表示希望的消息已经存在。假定用信号量 S 表示某条消息，进程可以通过调用 P 操作测试消息是否达到，调用 V 操作通知消息已经准备好。

3. 高级通信原语

P、V 操作是用来协调进程间关系的，编程较困难、效率低，通信对用户不透明，生产者每次只能向缓冲区放一个消息，消费者只能从缓冲区中取一个消息。所以交换的信息量多时要引入高级通信原语。进程高级通信的类型主要有以下几种。

(1) 共享存储系统。相互通信的进程共享某些数据结构或存储区，以实现进程之间的通信。

(2) 消息传递系统。进程间的数据交换以消息为单位，程序员直接利用系统提供的一组通信命令(原语)来实现通信，如 Send(A)、Receive(A)。

(3) 管道通信。管道是指用于连接两个进程之间的一个打开的共享文件(pipe 文件)。向管道(共享文件)提供输入的发送进程(即写进程)，以字符流的形式将大量的数据送入管道；而接收进程可从管道的另一端接收大量的数据。由于通信时采用管道，所以叫管道通信。

4.2.1.4 管程

管程是由一些共享数据、一组能为并发进程执行的作用在共享数据上的操作集合、初始代码以及存取权组成的。

采用这种方式管理共享资源可以借助数据结构及在其上实施操作的若干过程来进行，对共享资源的申请和释放可以通过过程在数据结构上的操作来实现。

管程提供了一种允许多进程安全有效地共享抽象数据类型的机制，管程实现同步机制的基础是“条件结构”。为实现进程的互斥和同步，必须定义一些条件变量。这些条件变量只能被 wait 和 signal 访问。利用管程可以解决生产者-消费者问题。

4.2.1.5 进程调度

在某些操作系统中，一个作业从提交到完成需要经历高、中、低 3 级调度。

(1) 高级调度。又称“长调度”“作业调度”或“接纳调度”。它决定处于输入池中的哪个后备作业可以调入主系统做好运行的准备，使其成为一个或一组就绪进程。系统中一个作业只需经过一次高级调度。

(2) 中级调度。又称“中程调度”或“对换调度”。它决定处于交换区中的就绪进程哪个可以调入主存，以便使其直接参与对 CPU 的竞争。在主存资源紧张时，为了将进程调入主存，必须将主存中处于阻塞状态的进程调至交换区，以便为调入进程腾出空间。

(3) 低级调度。又称“短程调度”或“进程调度”。它决定处于主存中的就绪进程哪个可以占用 CPU，是操作系统中最活跃、最重要的调度程序，对系统的影响很大。

1. 调度方式

调度方式是指当有更高优先级的进程到来时如何分配 CPU。调度方式分为可剥夺式和不可剥夺式两种。可剥夺式是指当有更高优先级的进程到来时，强行将正在运行的进程所占用的 CPU 分配给高优先级的进程；不可剥夺式是指当有更高优先级的进程到来时，必须等待正在运行的进程自动释放占用的 CPU，然后将 CPU 分配给高优先级的进程。

2. 进程调度算法

常用的进程调度算法有先来先服务、时间片轮转、优先级调度和多级反馈调度算法。

1) 先来先服务

先来先服务(FCFS)是按照作业提交或进程变为就绪状态的先后次序分配 CPU。即每当进入进程调度时，总是将就绪队列队首的进程投入运行。FCFS 主要用于宏观调度，其特点是比较有利于长作业，而不利于短作业；有利于 CPU 繁忙的作业，而不利于 I/O 繁忙的作业。

2) 时间片轮转

时间片轮转的基本思路是通过时间片轮转，提高进程并发性和响应时间，从而提高资源利用率。时间片轮转算法主要用于微观调度，其设计目标是提高资源利用率。

3) 优先级调度

优先级调度分为静态优先级和动态优先级两种。

(1) 静态优先级。进程的优先级是在创建时就已确定好了，直到进程终止都不会改变。

(2) 动态优先级。在创建进程时赋予一个优先级，在进程运行过程中还可以改变，以便获得更好的调度性能。

4) 多级反馈调度

多级反馈调度算法是在时间片轮转算法和优先级算法的基础上改进的。其优点是：照顾短进程，提高系统吞吐量，缩短平均周转时间；照顾 I/O 型进程以获得较好的 I/O 设备利用率和缩短响应时间；不必估计进程的执行时间，动态调节优先级。

优先级的确定需要考虑以下情况。

(1) I/O 型进程。让其进入最高优先级队列，以便能及时响应需要 I/O 交互的进程。通常执行一个小的时间片，在该时间片内，要求能处理完一次 I/O 请求的数据，然后转入阻塞队列。

(2) 计算型进程。每次都执行完时间片，进入更低级序列。最终采用最大时间片来执行，减少调度次数。

(3) I/O 次数不多而主要是 CPU 处理的进程。在 I/O 完成后，放回优先的 I/O 申请时离

开的队列，以免每次都回到最高优先级队列后再逐次下降。

(4) 为适应一个进程在不同时间段的运行特点，在 I/O 完成时，提高优先级；时间片用完时，降低优先级。

4.2.1.6 死锁

死锁是指两个以上的进程互相都因要求对方已经占有的资源，导致无法运行下去的现象。死锁是系统的一种出错状态，不仅浪费大量的系统资源，甚至会导致整个系统的崩溃，所以死锁是应该尽量预防和避免的。

1. 产生死锁的原因

产生死锁的原因是资源竞争及进程推进顺序非法。

2. 产生死锁的 4 个必要条件

产生死锁的 4 个必要条件如下。

(1) 互斥条件。进程对其要求的资源进行排他性控制，即一次只允许一个进程使用。

(2) 请求保持条件。零星地请求资源，即已获得部分资源后又请求资源被堵塞。

(3) 不可剥夺条件。进程已获得资源在未使用完之前不能被剥夺，只能在使用完时由自己释放。

(4) 环路条件。发生死锁时，在进程资源有向图中必构成环路，其中每个进程占有下一个进程申请的一个或多个资源。

3. 进程资源有向图

进程资源有向图由方框、圆圈和有向边 3 部分组成。其中，方框表示资源，圆圈表示进程。

请求资源：○→□，箭头由进程指向资源。

分配资源：○←□，箭头由资源指向进程。

4. 死锁的处理

下面介绍死锁的处理。

(1) 死锁的预防。根据产生死锁的 4 个必要条件，只要使其中之一不能成立，死锁就不会出现。为此，可以采取下列预防措施，即预先静态分配法和资源有序分配法。

(2) 死锁的避免。最著名的死锁避免算法是 Dijkstra 提出的银行家算法，其思想是：对于进程发出的每一个系统可以满足的资源请求命令加以检测，如果发现分配资源后，系统可能进入不安全状态，则不予分配；若分配资源后系统仍处于安全状态，则分配资源。与死锁预防策略相比提高了资源的利用率，但增加了系统的开销。

(3) 死锁的检测。这种方法对资源的分配如不加限制，即允许死锁发生。但系统定时地运行一个“死锁检测”程序，判断系统是否发生死锁，若检测到有死锁，则设法加以解除。

(4) 死锁的解除。检测到死锁发生后，常采用资源剥夺法和撤销进程法解除死锁。

4.2.1.7 线程

1. 线程的基本概念

线程是比进程更小的能独立运行的基本单位。在引入线程的操作系统中，线程是进程

中的一个实体，是系统独立分配和调度的基本单位。线程自己基本上不拥有资源，只拥有一点在运行中必不可少的资源(如程序计数器、一组寄存器和栈)，但它可与同属一个进程的其他线程共享该进程所占用的全部资源。一个线程可以创建和撤销另一个线程，同一个进程中的多个线程之间可以并发执行。线程也同样有就绪、等待和运行 3 种基本状态。

2. 线程的分类

线程的分类如下。

(1) 用户级线程。不依赖于内核，该类线程的创建、撤销和切换都不利用系统调用实现。

(2) 内核支持线程。依赖于内核，即无论是用户进程中的线程，还是系统中的线程，它们的创建、撤销和切换都利用系统调用实现。

(3) 同时实现了两种类型的线程。

3. 线程与进程的比较

下面介绍线程与进程的比较。

(1) 调度。将线程作为调度和分配的基本单位，进程作为拥有资源的基本单位。

(2) 并发性。不仅进程之间可并发执行，而且同一个进程中的多个线程之间也可并发执行。

(3) 拥有资源。进程是拥有资源的一个独立单位，线程不拥有系统资源，但可访问隶属于进程的资源。

(4) 系统开销。在创建或撤销进程时，由于系统都要为之分配和回收资源，导致系统的开销明显地大于创建或撤销线程时的开销。

4.2.2　典型例题分析

例 1　假设某分时系统采用简单时间片轮转法，当系统中的用户数为 *n*、时间片为 *q* 时，系统对每个用户的响应时间 *T*=<u>(23)</u>。(2013 年上半年试题 23)

(23) A. *n*　　B. *q*　　C. *nq*　　D. *n*+*q*

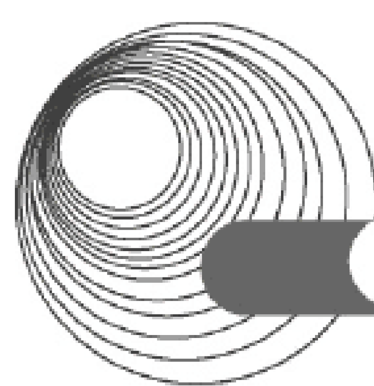
解析：在时间片轮转法中，系统将所有的就绪进程按先来先服务的原则排成一个队列，每次调度时，把 CPU 分配给队首进程，并令其执行一个时间片。当执行的时间片用完时，中断请求，将该程序送往就绪队列的队尾，并把处理机分配给新的队首进程，同时让它也执行一个时间片。这样就保证就绪队列中的所有进程在一给定的时间内均能获得一时间片的处理机执行时间。按照这样的思想，每个用户就是一个进程，这样每个用户的响应时间为 *nq*。

答案：C

例 2　在支持多线程的操作系统中，假设进程 P 创建了若干个线程，那么<u>(24)</u>是不能被这些线程共享的。(2013 年上半年试题 24)

(24) A. 该进程的代码段　　B. 该进程中打开的文件
C. 该进程的全局变量　　D. 该进程中某线程的栈指针

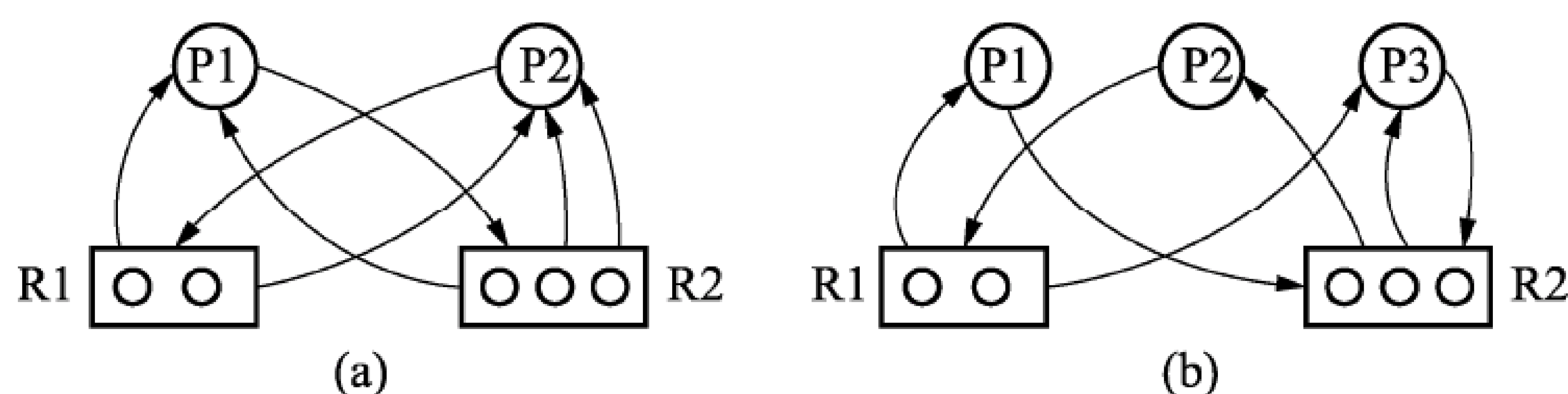
解析：线程共享的环境包括进程代码段、进程的公有数据(利用这些共享的数据，线程



很容易实现相互之间的通信)、进程打开的文件描述符、信号的处理器、进程的当前目录和进程用户 ID 与进程组 ID。但该线程中某线程的栈指针不能被这些线程所共享。

答案：D

例 3 进程资源图如图(a)和(b)所示，其中：图(a)中 (25) ；图(b)中 (26) 。(2013 年上半年试题 25、26)



(25) A. P1 是非阻塞节点，P2 是阻塞节点，所以该图不可以化简、是死锁的
B. P1、P2 都是阻塞节点，所以该图不可以化简、是死锁的
C. P1、P2 都是非阻塞节点，所以该图可以化简、是非死锁的
D. P1 是阻塞节点，P2 是非阻塞节点，所以该图不可以化简、是死锁的

(26) A. P1、P2、P3 都是非阻塞节点，该图可以化简、是非死锁的
B. P1、P2、P3 都是阻塞节点，该图不可以化简、是死锁的
C. P2 是阻塞节点，P1、P3 是非阻塞节点，该图可以化简、是非死锁的
D. P1、P2 是非阻塞节点，P3 是阻塞节点，该图不可以化简、是死锁的

解析：由图(a)可知，R1 资源共有两个，都被进程 P1 和 P2 所占用，R2 资源有 3 个，都被进程 P2 所占用，而此时进程 P1 在请求 R2 资源，P2 在请求 R1 资源，而 R1 和 R2 已经没有资源可用了，导致两个进程都进入了死锁状态。

由图(b)可知，R1 的两个资源已经分别被 P1 和 P3 所占用，R2 的 3 个资源已经占用了两个，当进程 P1 和 P3 请求资源 R2 的时候，无论分配给任意方都可以使两个进程满足所需的资源，从而可以化简，使得 P2 也能满足对资源 R1 的需求。因此 P1 和 P3 是非阻塞节点，P2 是阻塞节点，但最终可以化简。

答案：(25) B　　(26) C

例4 假设某计算机系统中资源R的可用数为6，系统中有3个进程竞争R，且每个进程都需要i个R，该系统可能会发生死锁的最小i值是 (24) 。若信号量S的当前值为-2，则R的可用数和等待R的进程数分别为 (25) 。(2014年上半年试题25)

(24) A. 1　　B. 2　　C. 3　　D. 4

(25) A. 0、0　　B. 0.1　　C. 1、0　　D. 0、2

解析：如果i=1，即每个进程都需要一个R，3个进程同时运行需要3个R，还剩3个R，不会发生死锁。如果i=2，即每个进程都需要两个R，3个进程同时运行需要6个R，而R的可用数正好为6，不会发生死锁。如果i=3，即每个进程都需要两个R，当3个进程分别占有两个R时，都需要再申请一个R资源才能正常运行，但此时已经没有R资源了，进程之间便出现了相互等待的状况，发生死锁。

信号量的值小于 0，表示没有可用的资源，其绝对值表示阻塞队列中等待该资源的进程数。

答案：(24)C　　(25)D

例 5　若某企业拥有的总资金数为 15，投资 4 个项目 P1、P2、P3、P4，各项目需要的最大资金数分别是 6、8、8、10，企业资金情况如图(a)所示。P1 新申请两个资金，P2 新申请一个资金，若企业资金管理处为项目 P1 和 P2 分配新申请的资金，则 P1、P2、P3、P4 尚需的资金数分别为__(23)__，假设 P1 已经还清所有投资款，企业资金使用情况如图(b)所示，那么企业的可用资金数为__(24)__。若在图(b)所示的情况下，企业资金管理处为 P2、P3、P4 各分配资金数 2、2、3，则分配后 P2、P3、P4 已用资金数分别为__(25)__。(2012 年上半年试题 23～25)

项目	最大资金	已用资金	尚需资金
P1	6	2	4
P2	8	3	5
P3	8	2	6
P4	10	3	7

(a)

项目	最大资金	已用资金	尚需资金
P1	—	—	—
P2	8	3	5
P3	8	2	6
P4	10	3	7

(b)

(23) A. 1、3、6、7，可用资金数为 0，故资金周转状态是不安全的
B. 2、5、6、7，可用资金数为 1，故资金周转状态是不安全的
C. 2、4、6、7，可用资金数为 2，故资金周转状态是安全的
D. 3、3、6、7，可用资金数为 2，故资金周转状态是安全的

(24) A. 4　　B. 5　　C. 6　　D. 7

(25) A. 3、2、3，尚需资金数分别为 5、6、7，故资金周转状态是安全的
B. 5、4、6，尚需资金数分别为 3、4、4，故资金周转状态是安全的
C. 3、2、3，尚需资金数分别为 5、6、7，故资金周转状态是不安全的
D. 5、4、6，尚需资金数分别为 3、4、4，故资金周转状态是不安全的

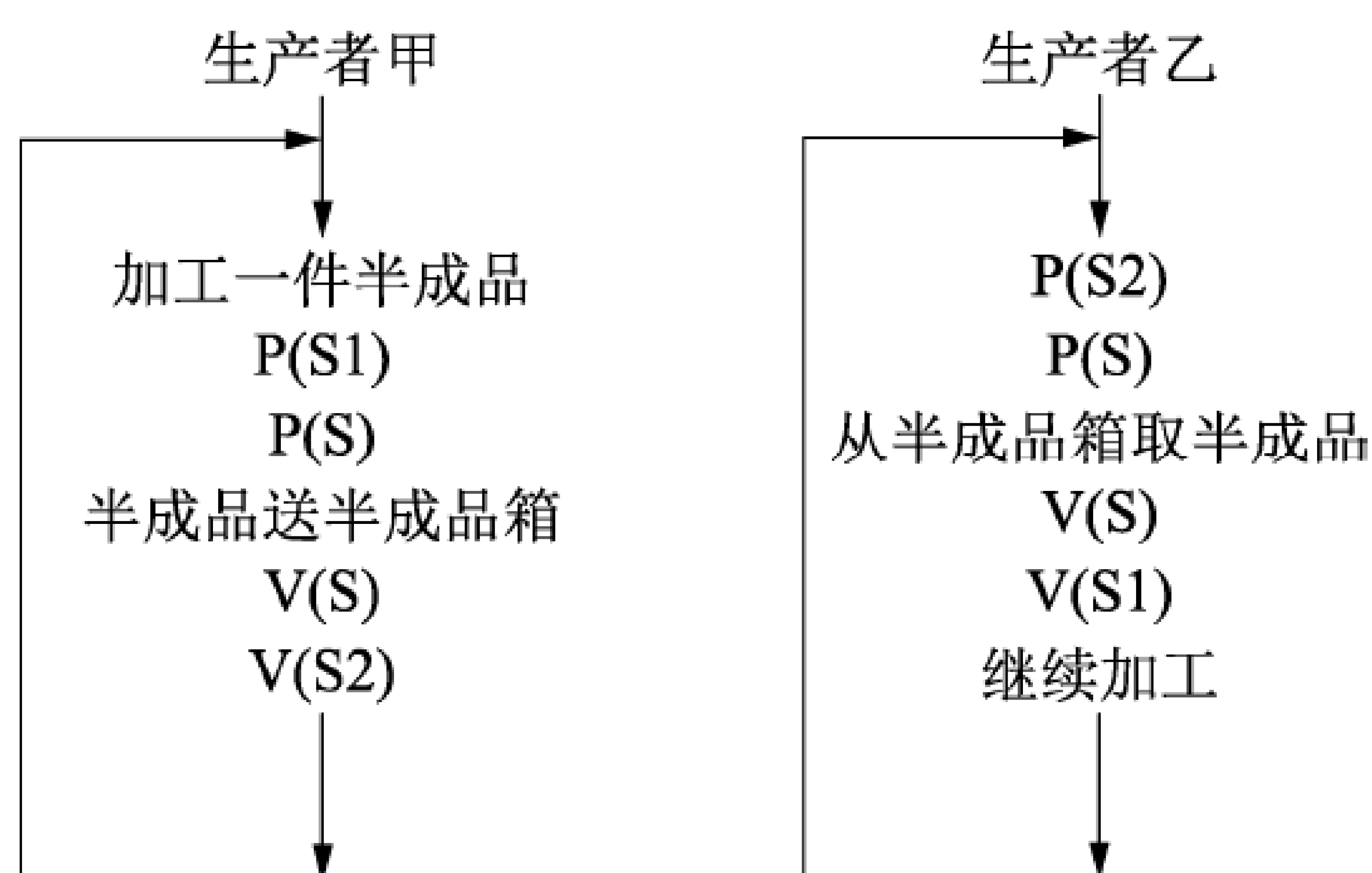
解析：由图(a)可知，之前已经给 4 个项目分配了 2+3+2+3=10 个资金。P1 新申请了两个资金，P2 新申请一个资金，则可用资金为 15−10−2−1=2，P1、P2、P3、P4 已用的资金数分别为 4、4、2、3，尚需的资金数为 2、3、6、7。故空(23)选 C。

由图(b)可知，已用资金数为 3+2+3=8，可用资金数为 15−8=7，故空(24)选 D。

在图(b)所示的情况下，企业资金管理处为 P2、P3、P4 各分配资金数 2、2、3，则 P2、P3、P4 已用的资金数分别为 3+2、2+2、3+3，即 5、4、6。尚需资金为 3、4、4，而此时总资金数为 0，故资金周转状态是不安全的。因此空(25)选 D。

答案：(23) C　(24) D　(25) D

例 6　某企业生产流水线 M 共有两位生产者，生产者甲不断地将其工序上加工的半成品放入半成品箱，生产者乙从半成品箱取出继续加工。假设半成品箱可存放 n 件半成品，采用 PV 操作实现生产者甲和生产者乙的同步可以设置 3 个信号量，即 S、S1 和 S2，其同步模型如下图所示。(2011 年下半年试题 22、23)



信号量 S 是一个互斥信号量，初值为 (22) ；S1、S2 的初值分别为 (23) 。

(22) A. 0　　B. 1　　C. *n*　　D. 任意正整数

(23) A. *n*、0　　B. 0、*n*　　C. 1、*n*　　D. *n*、1

解析：P 操作表示申请一个资源，V 操作表示释放一个资源。P 操作的定义为：$S:=S-1$，若 S=0，则执行 P 操作。

本题中，S 是一个互斥信号量，初值为 1，因为半成品箱是一个互斥资源，所以要进行互斥控制；S1 表示是否可以将半成品放入半成品箱，初值为 *n*；S2 表示半成品箱是否存有半成品，初值为 0。

答案：(22) B　　(23) A

例 7　系统中有 R 类资源 *m* 个，现有 *n* 个进程互斥使用。若每个进程对 *R* 资源的最大需求为 *w*，那么当 *m*、*n*、*w* 分别取下表中的值时，对于表中的①～⑥各情况， (25) 可能会发生死锁。若将这些情况的 *m* 分别加上 (26) ，则系统不会发生死锁。(2011 年上半年试题 25、26)

	①	②	③	④	⑤	⑥
m	3	3	5	5	6	6
n	2	3	2	3	3	4
w	2	2	3	3	3	2

(25) A. ①、②、⑤　　B. ③、④、⑤　　C. ②、④、⑤　　D. ②、④、⑥

(26) A. 1、1、1　　B. 1、1、2　　C. 1、1、3　　D. 1、2、1

解析：对于第①种情况，资源数为 3，在最坏的情况下，两个进程对 *R* 资源的需求都为 2，有一个进程占用两个 *R* 资源，另一个进程占用一个 *R* 资源，占用两个 *R* 资源的进程运行完，释放资源，另一个便会获得所需的资源，不会出现死锁。

对于第②种情况，资源数为 3，在最坏的情况下，3 个进程对 *R* 资源的需求都为 2，每个进程都占用一个 *R* 资源，发生死锁。

对于第③种情况，资源数为 5，在最坏的情况下，两个进程对 *R* 资源的需求都为 3，一个进程占用 3 个 *R* 资源，一个进程占用两个 *R* 资源，不会出现死锁。

对于第④种情况，资源数为 5，在最坏的情况下，3 个进程对 *R* 资源的需求都为 3，如果 3 个进程占用的资源数分别为 2、2、1，就会出现死锁。

对于第⑤种情况，资源数为 6，在最坏的情况下，3 个进程对 *R* 资源的需求都为 3，如

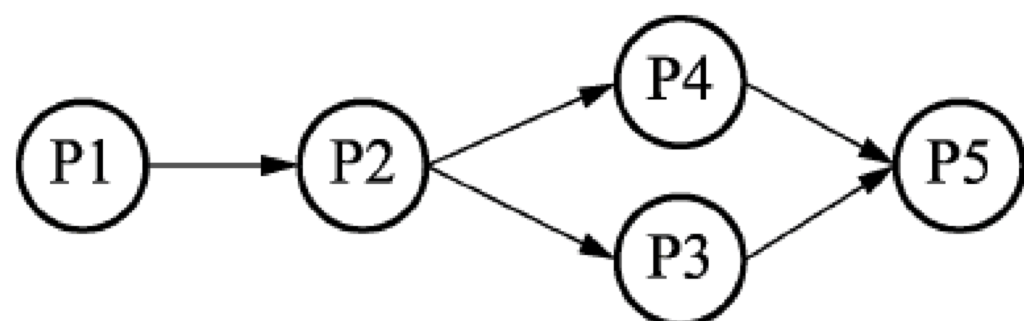
果 3 个进程占用的资源数都为 2，就会出现死锁。

对于第⑥种情况，资源数为 6，在最坏的情况下，4 个进程对 *R* 资源的需求都为 2，其中两个进程占用两个资源，两个进程占用一个资源，不会死锁。

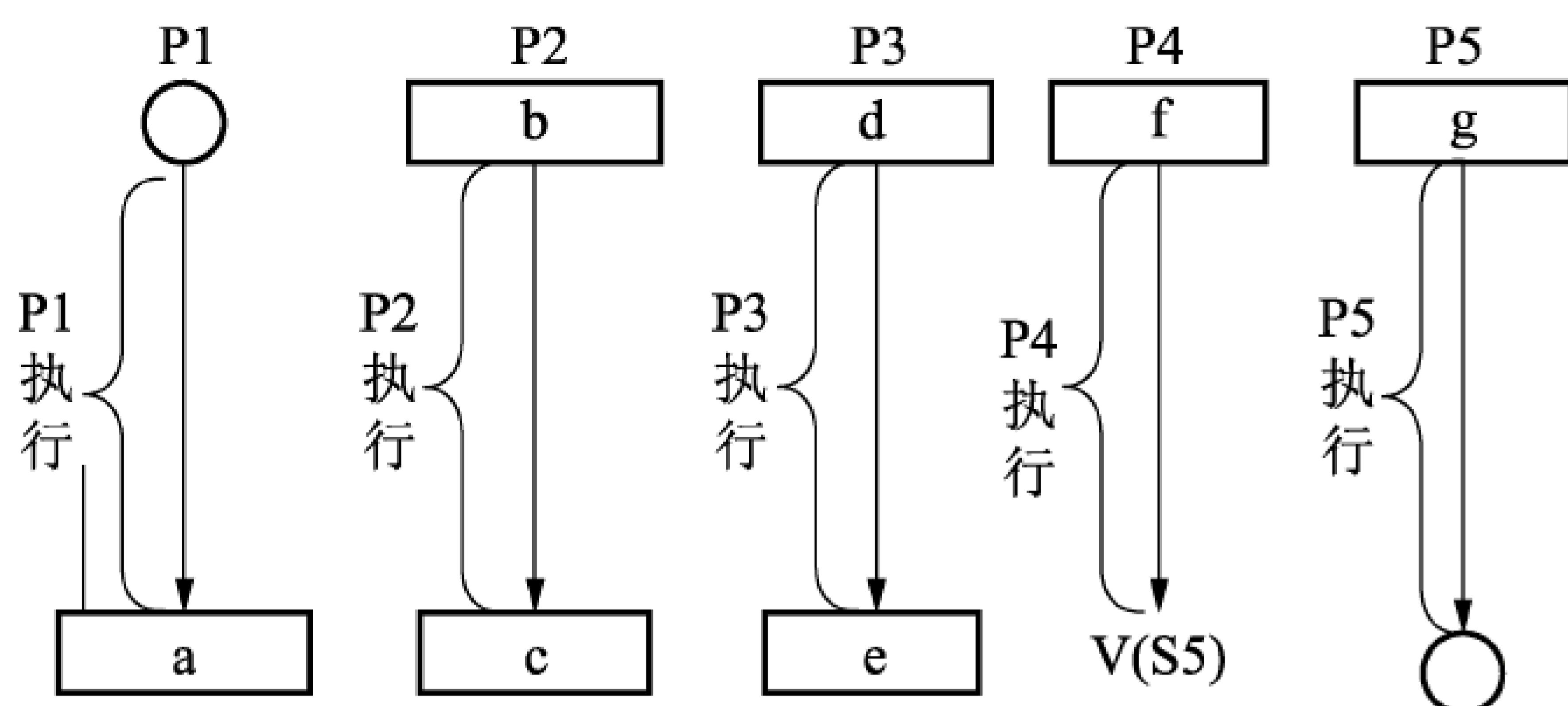
总结一下，可以发现，当 $n \times w - m > n$ 时，系统就不会发生死锁。因此，对于第②、⑤种情况，只要 *m* 的资源加 1，就不会发生死锁；对于第④种情况，*m* 资源需要加 2，系统就不会发生死锁。

答案：(25) C　　(26) D

例 8　进程 P1、P2、P3、P4 和 P5 的前趋图如下所示：



若用 PV 操作控制进程 P1、P2、P3、P4 、P5 并发执行的过程，则需要设置 5 个信号量 S1、S2、S3、S4 和 S5，且信号量 S1～S5 的初值都等于零。下图中 a、b 和 c 处应分别填写 __(23)__；d 和 e 处应分别填写 __(24)__，f 和 g 处应分别填写 __(25)__。(2015 年上半年试题 23、24、25)



(23) A. V(S1)、P(S1)和 V(S2)V(S3)　　B. P(S1)、V (S1)和 V(S2)V(S3)
　　 C. V(S1)、V(S2)和 P(S1)V(S3)　　D. P(S1)、V(S2)和 V (S1)V(S3)

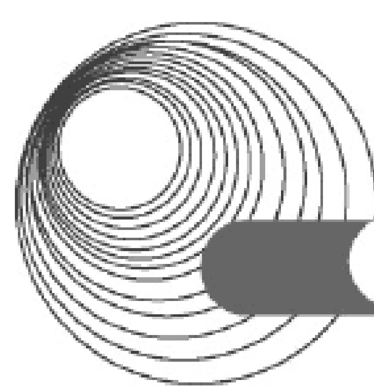
(24) A. V(S2)和 P(S4)　　B. P(S2)和 V(S4)
　　 C. P(S2)和 P(S4)　　D. V(S2)和 V(S4)

(25) A. P(S3)和 V(S4)V(S5)　　B. V(S3)和 P(S4)P(S5)
　　 C. P(S3)和 P(S4)P(S5)　　D. V(S3)和 V(S4)V(S5)

解析：进程通过 P 操作测试消息是否到达，通过 V 操作通知消息已经准备好。当 P1 执行完，通过 V(S1)通知 P2 已经准备好；P2 在执行之前，要通过 P(S2)操作测试消息是否到达，P2 执行完成之后，通过 V(S2)V(S3)通知 P3、P4 已经准备好；P3 在执行之前，通过 P(S2)测试消息是否到达，P3 执行完成之后，通过 V(S4)通知 P5 已经准备好；P4 在执行之前，通过 P(S3)测试消息是否到达，P4 执行完成之后，通过 V(S5)通知 P5 已经准备好；P5 执行之前，通过 P(S4)P(S5)测试消息是否到达。

答案：(23)A　(24)B　(25) C

例 9　如果系统采用信箱通信方式，当进程调用 Send 原语被设置成“等信箱”状态时，其原因是 __(23)__。(2010 年上半年试题 23)



(23) A．指定的信箱不存在　　　　　　B．调用时没有设置参数

C．指定的信箱中无信件　　　　　　D．指定的信箱中存满了信件

解析：为了实现进程间的通信，可以设立一个通信机构——信箱，以发送信件以及接收回答信件为进程间通信的基本方式。采用信箱通信的最大好处是，发送方和接收方不必直接建立联系，没有处理时间上的限制。发送方可以在任何时间发信，接收方也可以在任何时间收信。为了实现信箱通信，必须提供相应的原语，如创建信箱原语、撤销信箱原语、发送信件原语和接收信件原语等。Send 原语是发送原语，当进程调用 Send 原语被设置成“等信箱”状态时，意味着指定的信箱存满了信件，无可用空间。

答案：D

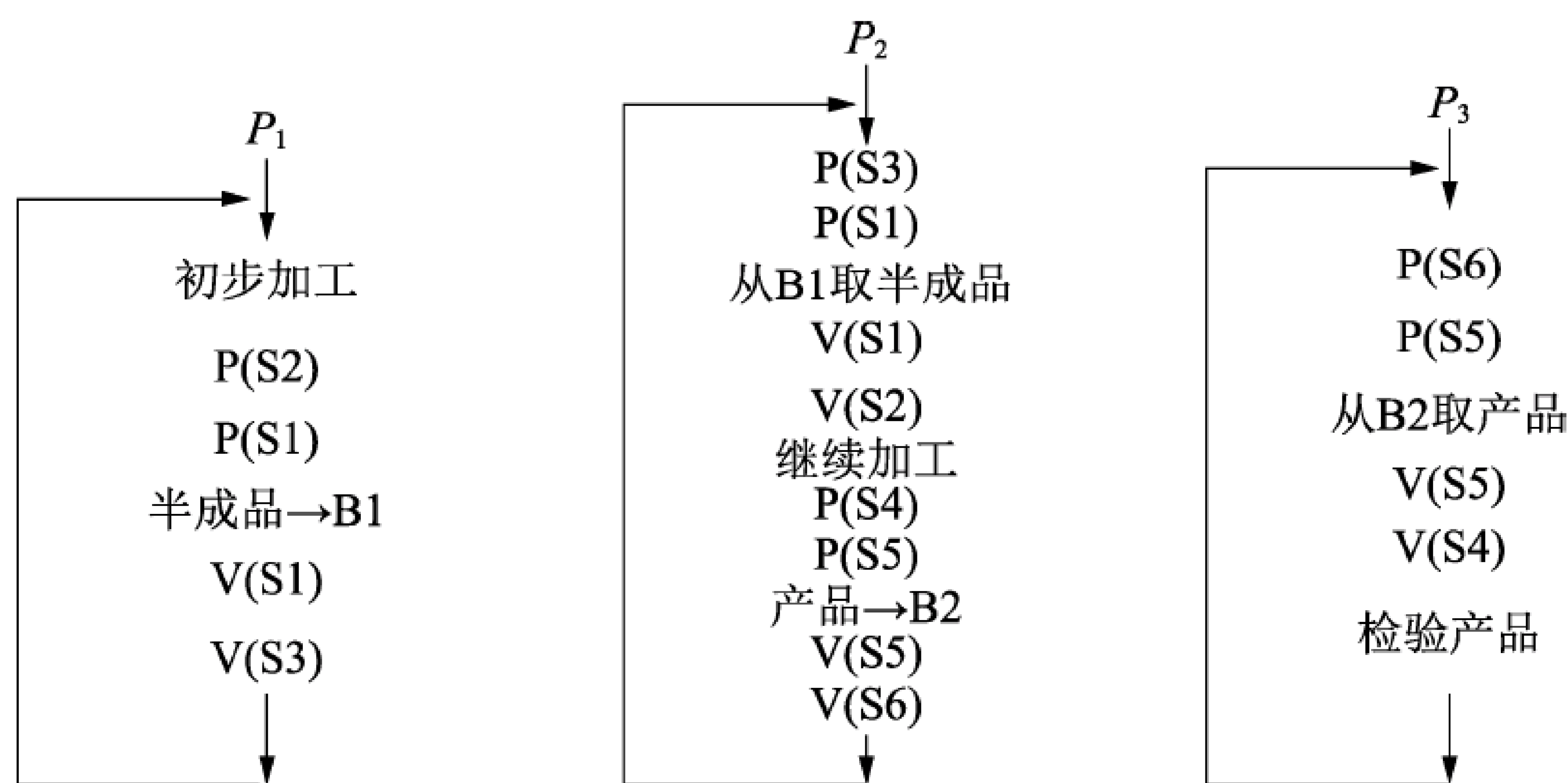
例 10　假设系统采用 PV 操作实现进程同步与互斥，若 n 个进程共享两台打印机，那么信号量 S 的取值范围为<u>(23)</u>。(2014 年下半年试题 23)

(23) A．-2~n　　　B．-(n-1)~1　　　C．-(n-1)~2　　　D．-(n-2)~2

解析：信号量初值等于资源数量，即为 2，由于同时最多有两个进程访问打印机，其余进程必须处理等待状态，故 S 的最小值为-(n-2)。

答案：(23) D

例 11　某企业的生产流水线上有两名工人 P1 和 P2，一名检验员 P3。P1 将初步加工的半成品放入半成品箱 B1；P2 从半成品箱 B1 取出继续加工，加工好的产品放入成品箱 B2；P3 从成品箱 B2 取出产品校验。假设 B1 可存放 n 件半成品，B2 可存放 m 件产品，并设置 6 个信号量 S1、S2、S3、S4、S5 和 S6，且 S3 和 S6 的初值都为 0。采用 PV 操作实现 P1、P2 和 P3 的同步模型如下图所示，则信号量 S1 和 S5 <u>(23)</u>；S2、S4 的初值分别为 <u>(24)</u>。(2015 年下半年试题 23、24)



(23) A．分别为同步信号量和互斥信号量，初值分别为 0 和 1

B．都是同步信号量，其初值分别为 0 和 0

C．都是互斥信号量，其初值分别为 1 和 1

D．都是互斥信号量，其初值分别为 0 和 1

(24) A．n、0　　　B．m、0　　　C．m、n　　　D．n、m

解析：从题目的描述可以了解到整个流程由 3 名不同的工人，按 P1→P2→P3 的顺序进行处理，这样要达到协作关系，需要使用同步信号量。同时由于 P1 的处理结果会存到 B1 中，P2 再从 B1 取内容，在此 B1 不能同时既进行存操作，也进行取操作，这就涉及互斥。

结合模型图可以看出：S1 信号量是互斥信号量，它确保 B1 的使用是互斥使用；S5 信号量针对 B2 起到同样的作用。

S2 与 S4 是同步信号量，S2 在 P1 开始放入半成品时执行 P 操作，代表资源占用，而在 P2 取出产品时执行 V 操作，代表资源释放，这说明 S2 对应的资源是 B1 的容量 n。同理 S4 对应 B2 的容量 m。

答案：(23) C　(24) D

例 12　设系统中有 *R* 类资源 *m* 个，现有 *n* 个进程互斥使用。若每个进程对 *R* 资源的最大需求为 *w*，那么当 *m*、*n*、*w* 取表中的值时，对于下表中的 *a*～*e* 五种情况，(26)两种情况可能会发生死锁。对于这两种情况，若将(27)，则不会发生死锁。(2009 年上半年试题 26、27)

	a	*b*	*c*	*d*	*e*
m	2	2	2	4	4
n	1	2	2	3	3
w	2	1	2	2	3

(26) A．*a* 和 *b*　　B．*b* 和 *c*　　C．*c* 和 *d*　　D．*c* 和 *e*

(27) A．*n* 加 1 或 *w* 加 1　　B．*m* 加 1 或 *w* 减 1

C．*m* 减 1 或 *w* 加 1　　D．*m* 减 1 或 *w* 减 1

解析：第 26 题，对于 *a*，有 *R* 类资源两个，只有一个进程，它需要两个该类资源即可完成。故不会发生死锁。对于 *b*，考虑最坏的情况，*R* 类资源有两个，两个进程各分得一个 *R* 类资源，能顺利做完。不会发生死锁。对于 *c*，同样考虑最坏的情况，两个进程各分得一个 *R* 类资源，由于这两个进程需要两个该类资源才能做完，并且已经没有多余的 *R* 类资源，故发生死锁。对于 *d*，考虑最坏的情况，*R* 类资源有 4 个，3 个进程各分得一个 *R* 类资源，剩下的一个资源任意分配各 3 个进程中的一个，使其顺利做完，然后释放该进程手中的资源，使其他进程也能顺利做完。故不会发生死锁。对于 *e*，考虑最坏情况，同样 3 个进程各分得一个资源，剩下一个资源分配给任意一个进程都不能使其做完，因为进程做完需要 3 个该类资源。故会发生死锁。

第 27 题在第 26 题分析的基础上，对于 *c* 和 *e*，资源数加 1 或者进程做完所需的最大资源数减 1 都能使其顺利完成，不会发生死锁。

答案：(26) D　(27) B

4.2.3　同步练习

1．假设系统中有三类互斥资源 R_1、R_2 和 R_3，可用资源数分别为 8、7 和 4。在 T_0 时刻系统中有 P_1、P_2、P_3、P_4 和 P_5 这 5 个进程，这些进程对资源的最大需求量和已分配资源数如表所示。在 T_0 时刻系统剩余的可用资源数分别为(1)。如果进程按(2)序列执行，那么系统状态是安全的。

资源 / 进程	最大需求量 R_1	R_2	R_3	已分配资源数 R_1	R_2	R_3
P_1	6	4	2	1	1	1
P_2	2	2	2	2	1	1
P_3	8	1	1	2	1	0
P_4	2	2	1	1	2	1
P_5	3	4	2	1	1	1

(1) A．0、1和0　B．0、1和1　C．1、1和0　D．1、1和1

(2) A．$P_1 \to P_2 \to P_4 \to P_5 \to P_3$　B．$P_2 \to P_1 \to P_4 \to P_5 \to P_3$

C．$P_4 \to P_2 \to P_1 \to P_5 \to P_3$　D．$P_4 \to P_2 \to P_5 \to P_1 \to P_3$

2．为了解决进程间的同步和互斥问题，通常采用一种称为__(1)__机制的方法。若系统中有5个进程共享若干个资源R，每个进程都需要4个资源R，那么使系统不发生死锁的资源R的最少数目是__(2)__。

(1) A．调度　B．信号量　C．分派　D．通信

(2) A．20　B．18　C．16　D．15

3．假设系统中有四类互斥资源R_1、R_2、R_3和R_4，可用资源数分别为9、6、3和3。在T_0时刻系统中有P_1、P_2、P_3和P_4等4个进程，这些进程对资源的最大需求量和已分配资源数如表所示。在T_0时刻系统剩余的可用资源数分别为__(1)__。如果P_1、P_2、P_3和P_4进程按__(2)__序列执行，那么系统状态是安全的。

(1) A．2、1、0和1　B．3、1、0和0

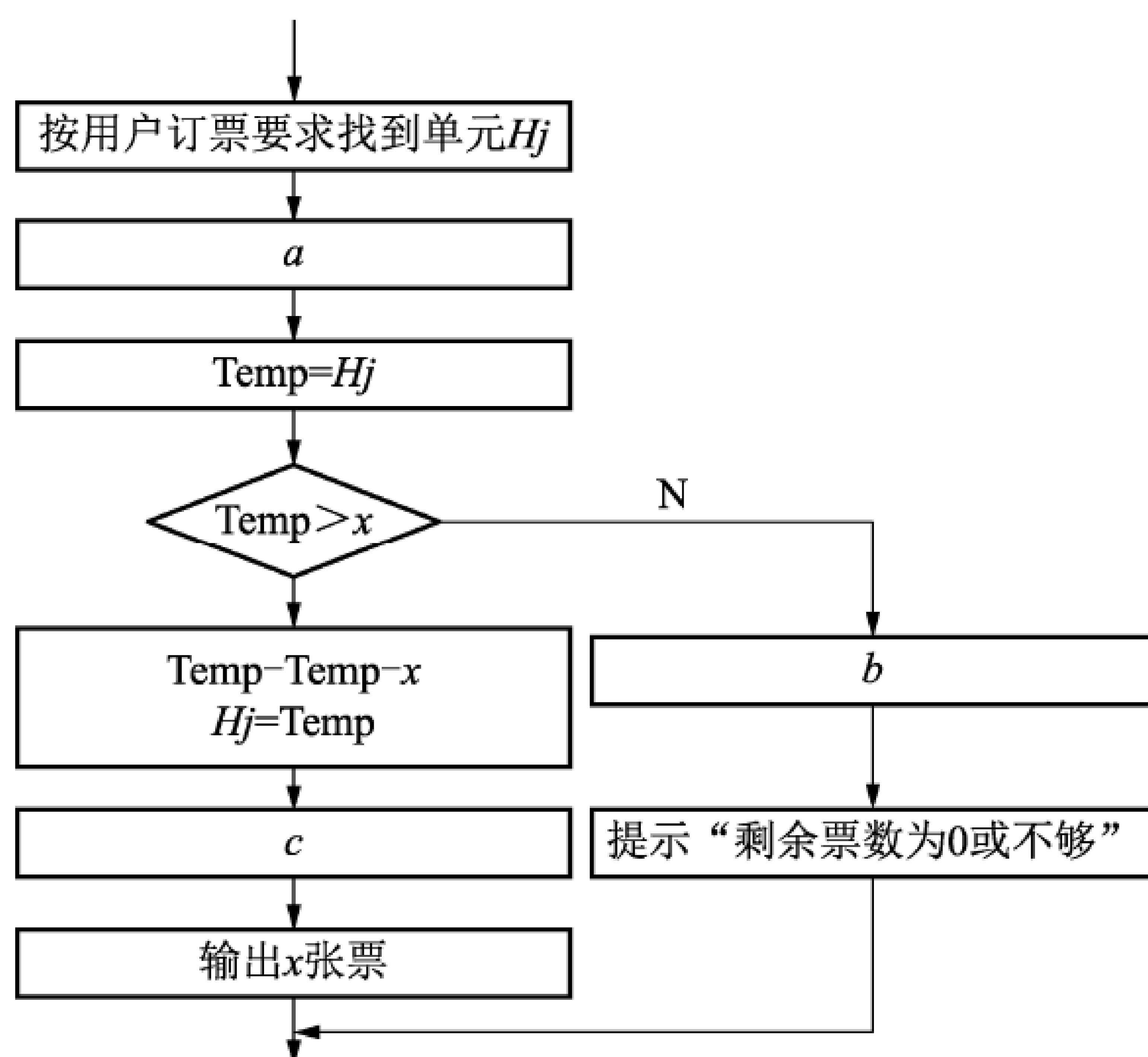
C．3、1、1和1　D．3、0、1和1

(2) A．$P_1 \to P_2 \to P_4 \to P_3$　B．$P_2 \to P_1 \to P_4 \to P_3$

C．$P_3 \to P_4 \to P_1 \to P_2$　D．$P_4 \to P_2 \to P_1 \to P_3$

资源 / 进程	最大需求量 R_1	R_2	R_3	R_4	已分配资源数 R_1	R_2	R_3	R_4
P_1	6	4	2	1	1	1	1	1
P_2	2	2	2	1	2	1	1	1
P_3	8	1	1	1	2	1	0	0
P_4	2	2	1	1	1	2	1	1

4．某火车票销售系统有n个售票点，该系统为每个售票点创建一个进程$P_i(i=1,2,\cdots,n)$。假设$H_j(j=1,2,\cdots,m)$单元存放某日某车次的剩余票数，Temp为P_i进程的临时工作单元，x为某用户的订票张数。初始化时系统应将信号量S赋值为__(1)__。P_i进程的工作流程如下图所示，若用P操作和V操作实现进程间的同步与互斥，则下图中a、b和c应分别填入__(2)__。



(1)　A．0　　B．1　　C．2　　D．3

(2)　A．P(*S*)、V(*S*)和 V(*S*)　　B．P(*S*)、P(*S*)和 V(*S*)

C．V(*S*)、P(*S*)和 P(*S*)　　D．V(*S*)、V(*S*)和 P(*S*)

5．某系统中有 4 种互斥资源 R_1、R_2、R_3 和 R_4，可用资源数分别为 3、5、6 和 8。假设在 T_0 时刻有 P_1、P_2、P_3 和 P_4 四个进程，并且这些进程对资源的最大需求量和已分配资源数如表所示，那么在 T_0 时刻系统中 R_1、R_2、R_3 和 R_4 的剩余资源数分别为(1)。如果从 T_0 时刻开始进程按(2)顺序逐个调度执行，那么系统状态是安全的。

(1)　A．3、5、6 和 8　　B．3、4、2 和 2

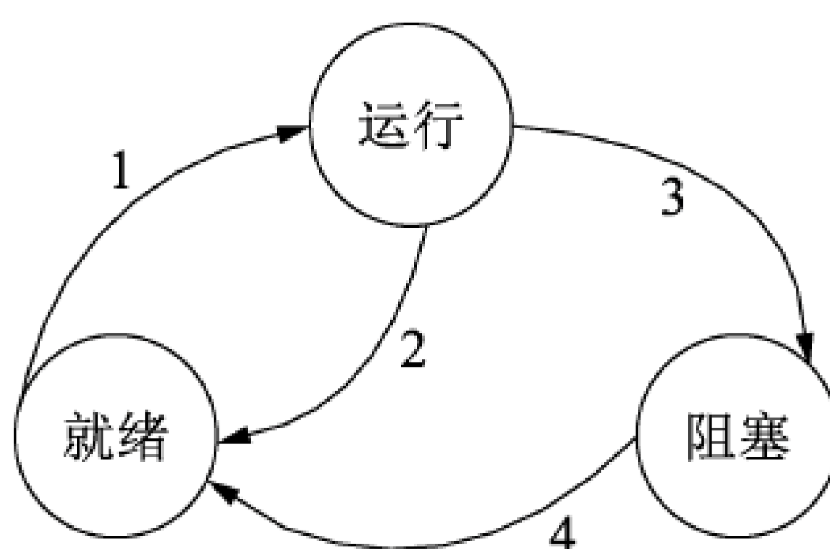
C．0、1、2 和 1　　D．0、1、0 和 1

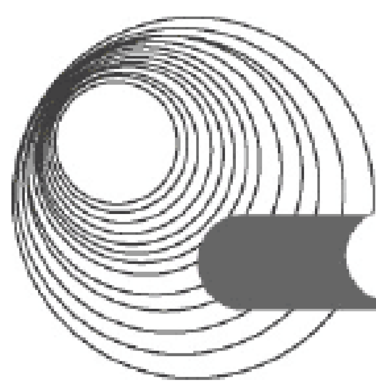
(2)　A．$P_1 \to P_2 \to P_4 \to P_3$　　B．$P_2 \to P_1 \to P_4 \to P_3$

C．$P_3 \to P_2 \to P_1 \to P_4$　　D．$P_4 \to P_2 \to P_3 \to P_1$

资源 进程	最大需求量				已分配资源数			
	R_1	R_2	R_3	R_4	R_1	R_2	R_3	R_4
P_1	1	2	3	6	1	1	2	4
P_2	1	1	2	2	0	1	2	2
P_3	1	2	1	1	1	1	1	0
P_4	1	1	2	3	1	1	1	1

6．某系统的进程状态转换如图所示，图中 1、2、3 和 4 分别表示引起状态转换的不同原因，原因 4 表示(1)；一个进程状态转换会引起另一个进程状态转换的是(2)。





(1) A．就绪进程被调度　　B．运行进程执行了 P 操作
　　C．发生了阻塞进程等待的事件　　D．运行进程的时间片到了

(2) A．1→2　　B．2→1　　C．3→2　　D．2→4

7．在一个单 CPU 的计算机系统中，采用可剥夺式(也称为抢占式)优先级的进程调度方案，且所有任务都可以并行使用 I/O 设备。表列出了 3 个任务 T_1、T_2、T_3 的优先级、独立运行时占用 CPU 和 I/O 设备的时间。如果操作系统的开销忽略不计，这 3 个任务从同时启动到全部结束的总时间为__(1)__ms，CPU 的空闲时间共有__(2)__ms。

任务	优先级	每个任务独立运行时所需的时间
T_1	高	对每个任务： 占用 CPU 10ms，I/O 13ms，再占用 CPU 5ms
T_2	中	
T_3	低	

(1) A．28　　B．58　　C．61　　D．64

(2) A．3　　B．5　　C．8　　D．13

4.2.4　同步练习参考答案

1．(1) C　(2) D　2．(1) B　(2) C
3．(1) B　(2) D　4．(1) B　(2) A
5．(1) D　(2) C　6．(1) C　(2) B
7．(1) B　(2) D

4.3　存 储 管 理

4.3.1　考点辅导

4.3.1.1　基本概念

1．*存储器的结构*

存储器的功能是保存数据，存储器的发展方向是高速度、大容量和小体积。一般存储器的结构有“寄存器-主存-外存”结构或“寄存器-缓存-主存-外存”结构。

(1) 虚拟地址。数据的存放地址是由符号决定的，故又称为符号名地址，或者称为名地址，而把源程序的地址空间叫做符号名地址空间或者名空间。它从 0 号单元开始编址，并顺序分配所有的符号名所对应的地址单元，所以它不是主存中的真实地址，故称为相对地址、程序地址、逻辑地址或虚拟地址。

(2) 地址空间。程序中由符号名组成的空间称为地址空间。源程序经过汇编或编译后再经过链接编辑程序加工形成程序的装配模块，即转换为相对地址编址的模块，它是以 0 为基址顺序进行编址的。相对地址也称为逻辑地址或虚拟地址，把程序中由相对地址组成

的空间叫做逻辑空间。相对地址空间通过地址重定位机构转换到绝对地址空间，绝对地址空间也叫物理地址空间。

(3) 存储空间。简单来说，逻辑地址空间(简称地址空间)是逻辑地址的集合，物理地址空间(简称存储空间)是物理地址的集合。

2. 地址重定位

地址重定位是指程序的逻辑地址被转换成主存的物理地址的过程。在可执行文件装入时需要解决可执行文件中地址(指令和数据)和主存地址的对应关系。由操作系统中的装入程序 Loader 和地址重定位机构来完成。地址重定位分为静态地址重定位和动态地址重定位。

(1) 静态地址重定位。它是指当用户程序被装入主存时已经实现了逻辑地址到物理地址的变换，在程序执行期间不再发生变化。其优点是：无需硬件地址转换机构的支持，只要求程序本身是可重定位的，它只对那些要修改的地址部分具有某种标识，由专门设计的程序来完成。

(2) 动态地址重定位。它是指在程序运行期间完成逻辑地址到物理地址的变换。其实现依赖于硬件地址变化机构，如基地址寄存器(BR)。其优点是：程序在执行期间可以被换入和换出主存，可以解决主存紧张的问题；可以在主存中移动，把主存中的碎片集中起来，以便充分利用空间；不必给程序分配连续的主存空间，以便较好地利用较小的主存块，实现共享。

4.3.1.2 存储管理方案

存储管理的主要目的是解决多个用户使用主存的问题，其存储管理方案主要包括分区存储管理、分页存储管理、分段存储管理、段页式存储管理及虚拟存储管理。

1. 固定分区

固定分区是一种静态分区方式，在系统生成时已将主存分成若干个分区，每个分区的大小可以不等，但分区大小固定不变，每个分区装一个且只能装一个作业。操作系统通过主存分配情况表管理主存。

固定分区的缺点是已分配区中存在未用空间，原因是程序或作业的大小不可能刚好等于分区的大小，故造成了空间的浪费。通常将已分配分区内的未用空间叫做零头或内碎片。

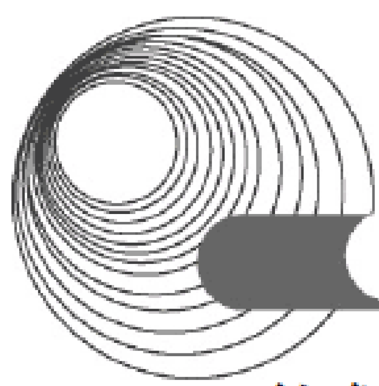
2. 可变分区

可变分区是一种动态分区方式，存储空间的划分是在作业装入时进行的，故分区的个数是可以变的，分区的大小刚好等于作业的大小。

引入可变分区方法，使主存分配有较大的灵活性，也提高了主存的利用率。但是可变分区会引起碎片的产生。解决碎片的方法是拼接(或称紧凑)，即向一个方向(如向低地址端)移动已分配的作业，使那些零散的小空闲区在另一方向连成一片。分区的拼接技术一方面要求能够对作业进行重定位，另一方面系统在拼接时要耗费较多的时间。

系统利用空闲分区表来管理主存中的空闲分区，请求和释放分区可以采用最佳适应算法、最差适应算法、首次适应算法和循环首次适应算法 4 种分配策略进行主存分配。

(1) 最佳适应算法。假设系统中有 X_1，X_2，X_3，…，X_i，…，X_n 个空闲区(自由区)，每当用户申请一个空间时，将从这 n 个空闲区中找到一个最接近用户需求的分区。这种算法



能保留较大的空白区。但缺点是空闲区不可能刚好等于用户要求的区，所以必然要将一个分区一分为二。随着系统不断地分配和释放空间，可能会使产生的小分区小到无法再继续分配，这样的无用小分区称为外碎片。

(2) 最差适应算法。接到主存申请时，系统总是将用户作业装入最大的空闲区。这种算法将一个最大的分区一分为二，所以剩下的空闲区通常也较大，不容易产生外碎片。

(3) 首次适应算法。每当用户作业申请一个空间时，系统总是从主存的低地址开始选择一个能装入作业的空白区。当用户释放空间时，该算法更易实现相邻的空白区合并。

(4) 循环首次适应算法。与首次适应算法的不同之处是，每次分配都是从刚分配的空闲区开始寻找一个能满足用户要求的空闲区。

3. 可重定位分区

可重定位分区是解决碎片问题的简单而又行之有效的方法。其基本思想是移动所有已分配好的分区，使之成为连续区域。由于移动分区是要付出代价的，所以通常是在用户请求空间得不到满足时进行。移动已分配的分区会导致地址发生变化，所以会产生地址重定位的问题。

分区保护的目的是防止未经核准的用户访问分区，常用以下两种保护方式。

(1) 上界/下界寄存器。采用上界/下界寄存器保护法时，上界寄存器中存放的是作业的装入地址，下界寄存器中存放的是作业的结束地址，形成的物理地址必须满足

上界寄存器≤物理地址≤下界寄存器

(2) 基址寄存器和限长寄存器。基址寄存器用来存放用户程序在主存的起始地址，限长寄存器用来存放用户程序的长度，形成的物理地址必须满足

基址寄存器≤物理地址<基址寄存器+限长寄存器

4.3.1.3 分页存储管理

1. 纯分页存储管理

1) 分页原理

系统将进程的地址空间划分成若干个大小相等的区域，称为页。同样地，将主存空间划分成与页相同大小的若干物理块，称为块或页框。在为进程分配主存时，将进程中若干页分别装入多个不相邻接的块中。

2) 地址结构

分页系统的地址结构如图 4-2 所示，它由两部分组成：前一部分为页号 *P*；后一部分为偏移量 *W*，即页内地址。图中的地址长度为 32 位，其中 0～11 位为页内地址(每页的大小为 4KB)，12～31 位为页号，所以允许地址空间的大小最多为 1MB 个页。



图 4-2 分页系统的地址结构

3) 页表

在将进程的每一页离散地分配到主存的多个物理块中后，系统应能保证在主存中找到每个页面所对应的物理块。为此，系统为每个进程建立了一张页面映射表，简称页表。每

个页在页表中占一个表项，记录该页在主存中对应的物理块号。进程在执行时，通过查找页表就可以找到每页所对应的物理块号。可见，页表的作用是实现从页号到物理块号的地址映射。

地址变换机构的基本任务是利用页表把用户程序中的逻辑地址变换成主存中的物理地址，实际上就是将用户程序中的页号变换成主存中的物理块号。为实现地址变换功能，在系统中设置页表寄存器，用来存放页表的地址和页表的长度。

2. 快表

在地址映射过程中，共需两次访问主存。第一次访问页表，得到数据的物理地址；第二次才是存取数据。为了提高访问主存的速度，可以采取两种方法：一种是在地址映射机制中增加一组高速寄存器保存页表，这需要大量的硬件开销，经济上不可行；另一种方法是在地址映射机制中增加一个小容量的联想寄存器(相联存储器)，它由一组高速寄存器组成，称为快表。快表用来存放当前访问最频繁的少数活动页的页号及相关信息。

在快表中，除了逻辑页号、物理页号对应外，还增加了几位：特征位表示该行是否为空；访问位表示该页是否被访问过，这是为了淘汰那些用得很少甚至不用的页面而设置的。

快表只存放当前进程最活跃的少数几页。当某一用户程序需要存取数据时，根据该数据所在逻辑页号在快表中找出对应的物理页号，然后与页内地址拼接成物理地址；如果在快表中没有相应的逻辑页号，则地址映射仍然通过主存中的页表进行，得到物理地址后需将该物理块号填到快表的空闲单元中。若无空闲单元，则根据淘汰算法淘汰某一行，再填入新得到的页号。实际上查找快表和查找主存页表是并行进行的，一旦在快表中找到相符的逻辑页号就停止查找主存页表。

4.3.1.4　分段存储管理

1. 基本原理

在分段存储管理方式中，作业的地址空间按程序自身的逻辑关系划分为若干个程序段，每个段是一组完整的逻辑信息。每个段都有自己的段名，且有一个段号。段号从 0 开始，每一段也从 0 开始编址，段内地址是连续的，各段长度是不等的。

分段系统的逻辑地址由段号(名)和段内地址两部分组成。在该地址结构中，允许一个作业最多有 64K 段，每个段的最大长度为 64KB。其地址结构如图 4-3 所示。

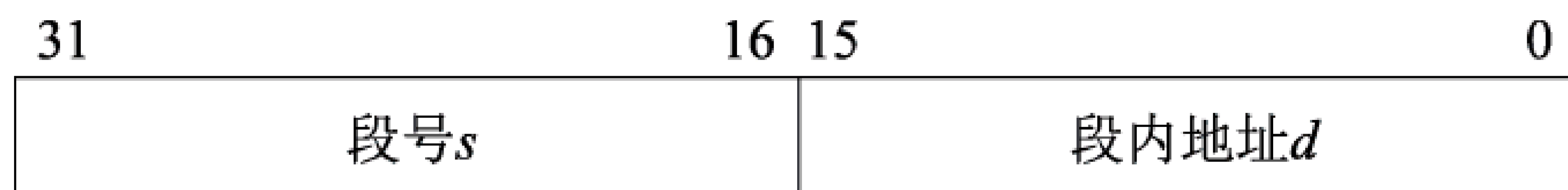
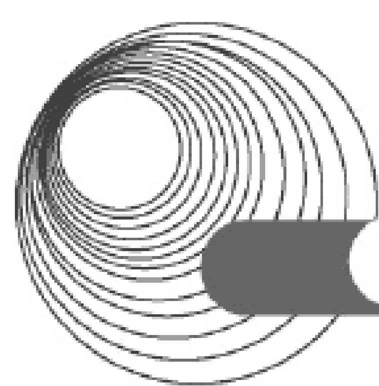


图 4-3　分段的地址结构

在分段式存储管理系统中，为每个段分配一个连续的分区，而进程中的各个段可以离散地分配到主存中不同的分区中。在系统中为每个进程建立一张段映射表，简称为“段表”。段表实现了从逻辑段到物理主存区的映射。

2. 分段系统的地址变换

为了实现从逻辑地址到物理地址的变换功能，系统中设置了段表寄存器，用于存放段表基址和段表长度。在进行地址变换时，系统对逻辑地址中的段号与段表长度进行比较。



段是信息的逻辑单位，因此分段系统的一个突出优点是易于实现段的共享，即允许若干个进程共享一个或多个段，而且对段的保护也十分简单。在分页系统中，虽然也能实现程序和数据的共享，但远不如分段系统来得方便。

4.3.1.5 段页式存储管理

段页式存储管理结合了段式存储管理和页式存储管理的优点，克服了两者的缺点。其基本原理是：先将整个主存划分成大小相同的存储块，将用户程序按程序的逻辑关系分为若干个段，并为每个段赋予一个段名，再将每个段划分成若干个页，以页架为单位离散分配。

在段页式系统中，其地址结构由段号、段内页号及页内地址三部分组成，如图 4-4 所示。因此，系统中需同时配置段表和页表。由于允许将一个段中的页进行离散分配，因而使段表的内容略有变化：它不再是段的主存基址和段长，而是页表始址和页表长度。

段号*s*	段内页号*p*	页内地址*w*

图 4-4　段页式管理的地址结构

在段页式系统中，为了便于实现地址变换，需要配置一个段表寄存器，存放段表基址和段表长度。

在进行地址变换时，首先利用段号 s，将它与段表长度 TL 进行比较。若 s<TL，表示未越界，于是利用段表基址和段号来求出该段对应的段表项在段表中的位置，从中得到该段的页表基址，并利用逻辑地址中的段内页号 p 来获得对应页的页表项位置，从中读出该页所在的物理块号 b，再利用物理块号 b 和页内地址构成物理地址。

4.3.1.6 虚拟存储管理

1. 虚拟存储器的引入

1) 局部性原理

局部性表现为时间局部性和空间局部性两类。

(1) 时间局部性。它是指最近被执行的指令可能再次被执行，最近被访问的存储空间很可能在不久的将来还要被访问。产生时间局部性的原因是在程序中存在着大量的循环操作。

(2) 空间局部性。它是指程序在一段时间内访问的地址可能集中在一定的范围内，其原因是程序的顺序执行。

2) 虚拟存储器的定义

根据局部性原理，一个作业在运行之前，没有必要把作业全部装入主存，而仅将当前要运行的那部分页面或段先装入主存启动运行，其余部分暂时留在磁盘上。

程序在运行时如果它所要访问的页(段)已调入主存，便可继续执行下去；但如果程序所要访问的页(段)尚未调入主存(称为缺页或缺段)，程序应利用操作系统所提供的请求调页(段)功能，将它们调入主存，以使进程能继续执行下去。

如果此时主存已满，无法再装入新的页(段)，则还要再利用页(段)的置换功能，将主存中暂时不用的页(段)调出至磁盘上，腾出足够的主存空间后，再将所要访问的页(段)调入主

存，使程序继续执行下去。这样，便可使一个大的用户程序在较小的主存空间中运行，也可使主存中同时装入更多的进程并发执行。从用户角度看，该系统所具有的主存容量比实际主存容量大得多，人们把这样的存储器称为虚拟存储器。

3)　虚拟存储器的功能

虚拟存储器具有请求调入功能和置换功能，能仅把作业的一部分装入主存便可运行作业，能从逻辑上对主存容量进行扩充。其逻辑容量由主存和外存容量之和以及 CPU 可寻址的范围来决定，其运行速度接近于主存速度。所以说，虚拟存储技术是一种性能非常优越的存储器管理技术，被广泛地应用于大、中、小型机和微型机中。

4)　虚拟存储器的实现

(1)　请求分页系统。在分页系统的基础上，增加了请求调页功能和页面置换功能所形成的页式虚拟存储系统。请求分页系统中的每个页表项包括页号、物理块号、状态位 *P*、访问字段 *A*、修改位 *M* 和外存地址。

(2)　请求分段系统。在分段系统的基础上，增加了请求调段和分段置换功能所形成的段式虚拟存储系统。它允许只装入若干段(而非全部段)的用户程序和数据，就可以启动运行，以后再通过调段功能和置换功能将不运行的段调出，同时调入将要运行的段，置换时以段为单位。

(3)　请求段页式系统。在段页式系统的基础上，增加了请求调页和页面置换功能形成的段页式虚拟存储系统。

5)　虚拟存储器的特征

虚拟存储器的特征包括离散性、多次性、对换性和虚拟性。

2.　请求分页管理的实现

请求分页系统是在纯分页系统的基础上，增加了请求调页功能、页面置换功能所形成的页式虚拟存储系统，是目前常用的一种虚拟存储方式。

请求分页的页表机制是在纯分页的页表机制上形成的，由于只将应用程序的一部分调入主存，还有一部分仍在磁盘上，故需在页表中再增加若干项，如状态位、访问字段和辅存地址等供程序(数据)在换进、换出时参考。

请求分页系统中的地址变换机构，是在分页系统的地址变换机构的基础上增加了某些功能，如产生和处理缺页中断，从主存中换出一页实现虚拟存储。

在请求分页系统中，每当所要访问的页面不在主存时，便要产生一个缺页中断，请求 OS 将所缺的页调入主存，这是由缺页中断机构完成的。缺页中断与一般中断的主要区别如下。

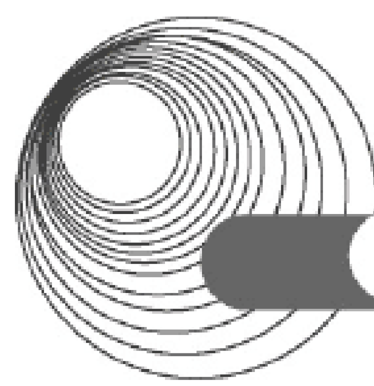
(1)　缺页中断在指令执行期间产生和处理中断信号，而一般中断是在一条指令执行完，下一条指令开始执行前检查和处理中断信号。

(2)　发生缺页中断时，返回到被中断指令的开始重新执行该指令，而一般中断返回到下一条指令执行。

(3)　一条指令在执行期间，可能会产生多次缺页中断。

3.　页面置换算法

请求分页系统的核心问题是选择合适的页面置换算法。常用的页面置换算法如下。



(1) 最佳(Optimal)置换算法。它是一种理想化的算法，性能最好，但在实际中难以实现，通常用来评价其他算法。

(2) 先进先出(FIFO)置换算法。该算法总是淘汰最先进入主存的页面，即选择在主存中驻留时间最久的页面予以淘汰。这是一种最直观，也是性能最差的算法，它有 Belady 异常现象，即如果对一个进程未分配它所要求的全部页面，有时就会出现分配的页面数增多但缺页率反而提高的异常现象。

(3) 最近最久未使用置换算法。该算法是选择最近最久未使用的页面予以淘汰，在实现时需要硬件的支持(寄存器或栈)。

(4) 最近未用置换算法。将最近一段时间未引用过的页面换出，是一种 LRU 的近似算法。

4. 工作集

工作集是指在某段时间间隔里进程实际要访问的页面集合。工作集理论认为，虽然程序只需要少量的几页就可以运行，但为了使程序能够有效地运行，较少地产生缺页，就必须使程序的工作集驻留在主存中。

4.3.2 典型例题分析

例 1 假设内存管理采用可变式分区分配方式，系统中有 5 个进程 P1～P5，且某一时刻内存使用情况如下图所示(图中空白处表示未使用分区)。此时，若 P5 进程运行完并释放其占有的空间，则释放后系统的空闲区数应 (27) ；造成这种情况的原因是 (28) 。(2013 年上半年试题 27、28)

分区号	进程
0	P1
1	P2
2	
3	P4
4	P3
5	
6	P5
7	

(27) A. 保持不变　B. 减 1　C. 加 1　D. 置零

(28) A. 无上邻空闲区，也无下邻空闲区
B. 有上邻空闲区，但无下邻空闲区
C. 有下邻空闲区，但无上邻空闲区
D. 有上邻空闲区，也有下邻空闲区

解析：可变分区是一种动态分区方式，在作业执行前并不建立分区，而是在处理作业过程中按需要建立分区。当 P5 运行完后释放空间时，发现其释放的空间上下方都有空闲区，故将两个空闲区与自身要释放的空闲区合并，从而形成一个空闲区，导致系统的空闲区数量上减 1。而造成这种现象的直接原因就是要释放的空闲区相邻的上下方也是空闲区。

答案：(27) B　(28) D

例 2　假设段页式存储管理系统中的地址结构如下图所示，则系统__(24)__。(2014 年下半年试题 24)

(24) A. 最多可有 2048 个段，每个段的大小均为 2048 个页，页的大小为 2K
　　B. 最多可有 2048 个段，每个段最大允许有 2048 个页，页的大小为 2K
　　C. 最多可有 1024 个段，每个段的大小均为 1024 个页，页的大小为 4K
　　D. 最多可有 1024 个段，每个段最大允许有 1024 个页，页的大小为 4K

31　　　　22	21　　　　12	11　　　　0
段号	页号	页内地址

解析： 段号数量 $2^{10}=1024$，段内最大页数 $2^{10}=1024$，页大小 $2^{12}=4096$B。

答案： D

例 3　对二维数组 $a[1..N,1..N]$中的一个元素 $a[i,j](1\leqslant i，j\leqslant N)$，存储在 $a[i,j]$之前的元素个数__(21)__。(2012 年上半年试题 21)

(21) A. 与按行存储或按列存储方式无关
　　B. 在 $i=j$ 时与按行存储或按列存储方式无关
　　C. 在按行存储方式下比按列存储方式下要多
　　D. 在按行存储方式下比按列存储方式下要少

解析： 存储在 $a[i,j]$之前的元素个数与按行存储或按列存储方式有关。按行存储时，存储在 $a[i,j]$之前的元素个数为$(i-1)*N+j-1=iN+j-N-1$；按列存储时，存储在 $a[i,j]$之前的元素个数为$(j-1)*N+i-1=jN+i-N-1$。很显然，$i<j$ 时，在按行存储方式下比按列存储方式下要多；$i>j$ 时，在按行存储方式下比按列存储方式下要少。

答案： B

例 4　若二维数组 arr[1..M, 1..N]的首地址为 base，数组元素按列存储且每个元素占用 K 个存储单元，则元素 arr[i, j]在该数组空间的地址为__(21)__。(2011 年下半年试题 21)

(21) A. base+$((i-1)*M+j-1)*K$　　　　B. base+$((i-1)*N+j-1)*K$
　　C. base+$((j-1)*M+i-1)*K$　　　　D. base+$((j-1)*N+i-1)*K$

解析： 数组 arr 共 M 行 N 列，下标均从 1 开始。元素 arr[i,j]在数组 arr 的第 i 行第 j 列，如果数组元素按列存储，则 1～$j-1$ 列共有$(j-1)*M$ 个元素，arr[i,j]之前共有$(j-1)*M+i-1$ 个元素，元素 arr[i, j]在该数组空间的地址为 base+$((j-1)*M+i-1)*K$。

答案： C

例 5　某程序设计语言规定在源程序中的数据都必须具有类型，然而，__(28)__并不是作出此规定的理由。(2011 年下半年试题 28)

(28) A. 为数据合理分配存储单元
　　B. 可以定义和使用动态数据结构
　　C. 可以规定数据对象的取值范围及能够进行的运算
　　D. 对参与表达式求值的数据对象可以进行合法性检查

解析： 若一种程序设计语言规定其程序中的数据必须具有类型，则有利于在翻译程序的过程中为数据合理分配存储单元、对参与表达式计算的数据对象进行检查、规定数据对象的取值范围及能够进行的运算。动态数据结构是在程序运行过程中动态地定义和使用，

在编译或解释过程中不会涉及，因此答案选择 B。

答案：B

例 6 某进程有 4 个页面，页号为 0~3，页面变换表及状态位、访问位和修改位的含义如下图所示。若系统给该进程分配了 3 个存储块，当访问前页面 1 不在内存时，淘汰表中页号为(26)的页面代价最小。(2015 年上半年试题 26)

页号	页帧号	状态位	访问位	修改位
0	6	1	1	1
1	—	0	0	0
2	3	1	1	1
3	2	1	1	0

状态位含义 {=0 不在内存；=1 在内存

访问位含义 {=0 未访问过；=1 访问过

修改位含义 {=0 未修改过；=1 修改过

(26) A. 0　　B. 1　　C. 2　　D. 3

解析：页面 1 不在内存，可以直接排除选项 B。在本题中，内存中的 3 个页面，都是刚刚被访问过的。所以不能以访问位作为判断标准，只能看修改位。修改位中，只有 3 号页未被修改，如果淘汰 3 号页，直接淘汰即可，没有额外的工作要做；如果淘汰 0 号或 2 号，则需要把修改的内容进行更新，这样会有额外的开销。

答案：D

例 7 某进程有 5 个页面，页号为 0～4，页面变换表如下表所示。表中状态位等于 0 和 1，分别表示页面不在内存或在内存。若系统给该进程分配了 3 个存储块，当访问的页面 3 不在内存时，应该淘汰表中页号为 (25) 的页面。假定页面大小为 4K，逻辑地址为十六进制 2C25H，该地址经过变换后，其物理地址应为十六进制 (26) 。(2010 年上半年试题 25、26)

页号	页帧号	状态位	访问位	修改位
0	3	1	1	0
1	—	0	0	0
2	4	1	1	1
3	—	0	0	0
4	1	1	1	1

(25) A. 0　　B. 1　　C. 2　　D. 4

(26) A. 2C25H　　B. 4096H　　C. 4C25H　　D. 8C25H

解析：页面变换表中状态位等于 0 和 1 分别表示页面不在内存或在内存，所以 0、2 和 4 号页面在内存。当访问的页面 3 不在内存时，系统应该首先淘汰未被访问的页面，因为根据程序的局部性原理，最近为被访问的页面下次被访问的概率更小；如果页面最近都被访问过，应该先淘汰未修改过的页面。因为未修改过的页面内存与辅存一致，故淘汰时无须写回辅存，使系统页面置换代价小。经上述分析，0、2 和 4 号页面都是最近被访问过的，但 2 和 4 号页面都被修改过而 0 号页面未修改过，故应该淘汰 0 号页面。

根据题意，页面大小为 4KB，逻辑地址为十六进制 2C25H，其页号为 2，页内地址为 C25H，查页表后可知页帧号(物理块号)为 4，该地址经过变换后，其物理地址应为页帧号 4

拼上页内地址 C25H，即十六进制 4C25H。

答案：(25) A　　(26) C

例 8　若用 256K×8bit 的存储器芯片，构成地址 40000000H 到 400FFFFFH 且按字节编址的内存区域，则需(6)片芯片。(2014 年上半年试题 6)

(6) A. 4　　B. 8　　C. 16　　D. 32

解析：内存区域从40000000H到400FFFFFH，占用的字节数为400FFFFFH-40000000H+1=100000H=1 0000 0000 0000 0000 0000B=220=1024K一片256K×8bit的存储器芯片的存储容量为256KB，需要的芯片数为1024÷256=4。

答案：A

例9　某计算机系统页面大小为4K，若进程的页面变换表如下所示，逻辑地址为十六进制1D16H。该地址经过变换后，其物理地址应为十六进制 (26) 。 (2014年上半年试题26)

页号	物理块号
0	1
1	3
2	4
3	6

(26) A.1024H　　B. 3D16H　　C. 4D16H　　D. 6D16H

解析：页面大小为4K，而4K=212，因此逻辑地址的低12位对应页内地址，高位对应页号。题目中逻辑地址为十六进制1D16H，一位十六进制数对应4位二进制数，3位十六进制数则对应12位二进制数，因此D16H为页内地址，页号为1。查页面变换表，页号1对应的物理块号为3，将物理块号与页内地址D16H拼接起来即可得到物理地址3D16H。

答案：B

4.3.3　同步练习

1.　某虚拟存储系统采用最近最少使用(LRU)页面淘汰算法。假定系统为每个作业分配 3 个页面的主存空间，其中一个页面用来存放程序。现有某作业的部分语句如下：

```
Var A: Array[1..128,1..128] OF integer;
I,j: integer;
FOR i:=1 to 128 DO
FOR j:=1 to 128 DO
A[i,j]:=0;
```

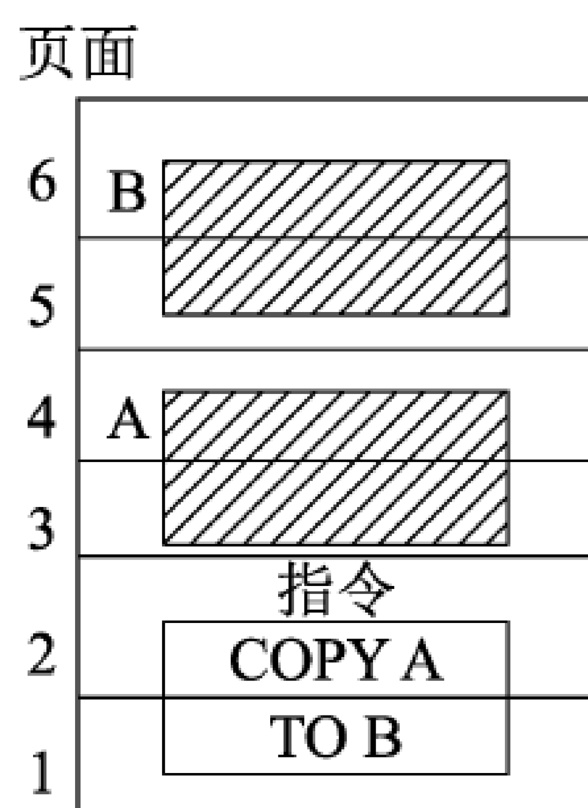
设每个页面可存放 128 个整数变量，变量 *i*、*j* 放在程序页中，矩阵 ***A*** 按行序存放。初始时，程序及变量 *i*、*j* 已在内存，其余两页为空。在上述程序片段执行过程中，共产生 (1) 次缺页中断。最后留在内存中的是矩阵 ***A*** 的最后 (2) 。

(1) A. 64　　B. 128　　C. 256　　D. 512

(2) A. 2 行　　B. 2 列　　C. 1 行　　D. 1 列

2.　在某计算机中，假设某程序的 6 个页面如下图所示，其中某指令 COPY A TO B 跨

两个页面，且源地址 A 和目标地址 B 所涉及的区域也跨两个页面。若地址为 A 和 B 的操作数均不在内存，计算机执行该 COPY 指令时，系统将产生(1)次缺页中断；若系统产生 3 次缺页中断，那么该程序应有(2)个页面在内存。

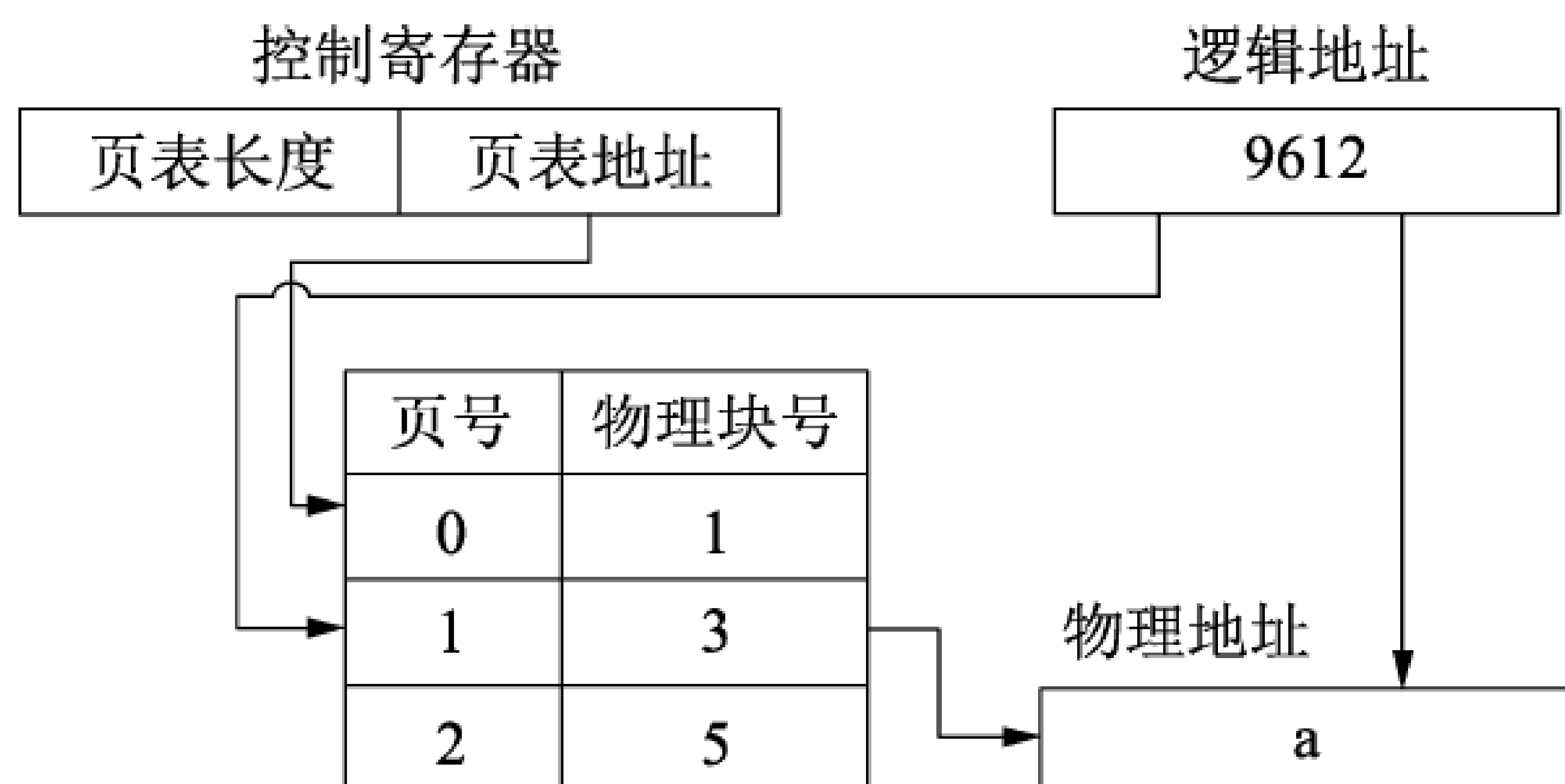


(1) A．2　　B．3　　C．4　　D．5

(2) A．2　　B．3　　C．4　　D．5

3．页式存储系统的逻辑地址是由页号和页内地址两部分组成的，地址变换过程如下图所示。假定页面的大小为 8KB，图中所示的十进制逻辑地址 9612 经过地址变换后，形成的物理地址 *a* 应为十进制_____。

A．42380　　B．25996　　C．9612　　D．8192



4．从下表关于操作系统存储管理方案 1、方案 2 和方案 3 的相关描述可以看出，它们分别对应_____存储管理方案。

A．固定分区、请求分页和覆盖　　B．覆盖、请求分页和固定分区

C．固定分区、覆盖和请求分页　　D．请求分页、覆盖和固定分区

方　案	说　明
1	在系统进行初始化的时候就已经将主存储空间划分成大小相等或不等的块，并且这些块的大小在此后是不可以改变的。系统将程序分配在连续的区域中
2	主存储空间和程序按固定大小单位进行分割，程序可以分配在不连续的区域中。该方案当一个作业的程序地址空间大于主存可以使用的空间时也可以执行
3	编程时必须划分程序模块和确定程序模块之间的调用关系，不存在调用关系的模块可以占用相同的主存区

5．主存按字节编址，地址从 A4000H～CBFFFH，共有_(1)_B。若用存储容量为 32K×8b 的存储器芯片构成该主存，至少需要_(2)_片。

(1) A．80K　　B．96K　　C．160K　　D．192K

(2) A．2　　B．5　　C．8　　D．10

6．容量为 64 块的 Cache 采用组相联方式映像，字块大小为 128B，每 4 块为一组。若主存容量为 4096 块，且以节编址，那么主存地址应为 (1) 位，主存区号应为 (2) 位。

(1) A．16　　B．17　　C．18　　D．19

(2) A．5　　B．6　　C．7　　D．8

7．虚拟存储管理系统的基础是程序的 (1) 理论，这个理论的基本含义是指程序执行时往往会不均匀地访问主存储器单元。根据这个理论，Denning 提出了工作集理论。工作集是进程运行时被频繁访问的页面集合。在进程运行时，如果它的工作集页面都在 (2) 内，则能够使该进程有效地运行；否则会出现频繁的页面调入/调出现象。

(1) A．全局性　　B．局部性　　C．时间全局性　　D．空间全局性

(2) A．主存储器　　B．虚拟存储器　　C．辅助存储器　　D．U 盘

8．假设某计算机系统的主存大小为 256KB，在某一时刻主存的使用情况如下表 1 所示。此时，若进程顺序请求 20KB、10KB 和 5KB 的存储空间，系统采用_____算法为进程依次分配主存，则分配后的主存情况如下表 2 所示。

A．最佳适应　　B．最差适应　　C 首次适应　　D．循环首次适应

表 1

起始地址	0K	20K	50K	90K	100K	105K	135K	160K	175K	195K	220K
状态	已用	未用	已用	已用	未用	已用	未用	已用	未用	未用	已用
容量	20KB	30KB	40KB	10KB	5KB	30KB	25KB	15KB	20KB	25KB	36KB

表 2

起始地址	0K	20K	40K	50K	90K	100K	105K	135K	145K	160K	175K	195K	200K	220K
状态	已用	已用	未用	已用	已用	未用	已用	已用	未用	已用	未用	已用	未用	已用
容量	20KB	20KB	10KB	40KB	10KB	5KB	30KB	10KB	15KB	15KB	20KB	5KB	20KB	36KB

9．使 Cache 命中率最高的替换算法是_____。

A．先进先出算法(FIFO)　　B．随机算法(RAND)

C．先进后出算法(FILO)　　D．替换最近最少使用的块算法(LRU)

4.3.4　同步练习参考答案

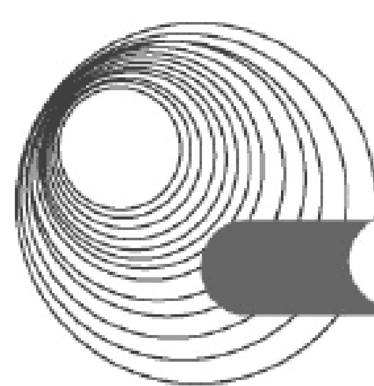
1．(1) B　(2) A　　2．(1) C　(2) B

3．B　　4．A

5．(1) C　(2) B　　6．(1) D　(2) B

7．(1) B　(2) A　　8．B

9．D



4.4 设备管理

4.4.1 考点辅导

4.4.1.1 设备管理概述

1. 设备的分类

1) 按数据组织分类

按数据组织分类，设备可分为块设备和字符设备。

(1) 块设备。它指以数据块为单位组织和传送数据的设备，如磁盘、磁带等，属于有结构设备。

(2) 字符设备。它指以单个字符为单位传送数据信息的设备，如终端、打印机等，属于无结构设备。

2) 按资源分配的角度分类

按资源分配的角度分类，设备可分为独占设备、共享设备和虚拟设备。

(1) 独占设备。在一段时间内只允许一个用户(进程)访问的设备。低速 I/O 设备一般是独占设备，如打印机、用户终端等。

(2) 共享设备。在一段时间内允许多个进程同时访问的设备。共享设备是可寻址的和可随机访问的设备。典型的共享设备是硬盘。共享设备不仅可以获得良好的设备利用率，而且是实现文件系统和数据库系统的物质基础。

(3) 虚拟设备。它指通过虚拟技术将一台独占设备变换为若干台供多个用户(进程)共享的逻辑设备。一般可利用假脱机技术(SPOOLing 技术)实现虚拟设备。

3) 按数据传输率分类

按数据传输率分类，设备可分为低速设备、中速设备和高速设备。

(1) 低速设备。它指传输速率为每秒钟几个字节到数百个字节的设备。典型的设备有键盘、鼠标、语音的输入设备等。

(2) 中速设备。它指传输速率在每秒钟数千字节至数万字节的设备。典型的设备有行式打印机、激光打印机等。

(3) 高速设备。它指传输速率在数百千个字节至数兆字节的设备。典型的设备有磁带机、磁盘机、光盘机等。

4) 其他分类方法

按输入输出对象分类，设备可分为人机通信设备和机机通信设备。

按是否可交互分类，设备可分为非交互设备和交互设备。

2. 设备管理的目标与任务

1) 设备管理的目标

设备管理的目标主要是如何提高设备的利用率，为用户提供方便、统一的界面。

2)　设备管理的任务

设备管理的任务是保证在多道程序环境下，当多个进程竞争使用设备时，按一定策略分配和管理各种设备，控制设备的各种操作，完成输入输出设备与主存之间的数据交换。

3)　设备管理的功能

设备管理的主要功能如下。

- 动态地掌握并记录设备的状态。
- 设备分配和释放。
- 缓冲区管理。
- 实现物理 I/O 设备的操作。
- 提供设备使用的用户接口。
- 设备的访问和控制。
- I/O 缓冲和调度。

4.4.1.2　I/O 软件

I/O 设备管理软件一般分为 4 层，即中断处理程序、设备驱动程序、与设备无关的系统软件和用户层 I/O 软件。至于一些具体分层时细节上的处理，是依赖于系统的，没有严格的划分，只要有利于设备独立于这一目标，就可以为了提高效率而设计不同的层次结构。

1. 中断处理程序

中断处理程序的作用是，当 I/O 操作结束时，唤醒驱动程序。中断处理过程如下。

(1)　CPU 检查响应中断的条件是否满足。
(2)　如果 CPU 响应中断，则 CPU 关中断，使其进入不可再次响应中断的状态。
(3)　保存并中断进程的现场。
(4)　分析中断原因，调用中断处理子程序。
(5)　执行中断处理子程序。
(6)　退出中断，恢复被中断进程的现场或调度新进程占据 CPU。
(7)　开中断，CPU 继续执行。

2. 设备驱动程序

设备驱动程序直接同硬件打交道，任务是接受来自与设备无关的上层软件的抽象请求，进行与设备相关的处理。最突出的特点是它与 I/O 设备的硬件结构联系密切。设备驱动程序中基本上是依赖于设备的代码，是操作系统底层中唯一知道各种输入输出设备的控制器细节及其用途的部分。

3. 与设备无关的系统软件

除了一些 I/O 软件与设备相关外，大部分软件是与设备无关的。设备驱动程序与设备无关的软件之间的界限如何划分随操作系统的不同而不同。具体划分原则取决于系统的设计者怎样权衡系统与设备的独立性、驱动程序的运行效率等诸多因素。

4. 用户层 I/O 软件

通常的系统调用包括 I/O 系统调用，是由库函数实现。所有库函数是设备管理 I/O 系统

的组成部分，通过这些库函数所做的工作主要是把系统调用时所用的参数放在合适的位置，由其他 I/O 过程去实现真正的操作。

另一种方法是采用 SPOOLing 系统，SPOOLing 系统是多道程序设计系统中模拟独占 I/O 设备完成假脱机的一种 I/O 技术。SPOOLing 技术不仅可适用于打印机这类输入输出设备，还可以应用到其他情况。

4.4.1.3 设备管理采用的相关缓冲技术

1. 通道技术

1) 通道

引入通道的目的是使数据的传输独立于 CPU，使 CPU 从繁重的 I/O 工作中解脱出来。设置通道后，CPU 只需向通道发出 I/O 命令，通道收到命令后，从主存中取出本次 I/O 要执行的通道程序并执行，仅当通道完成 I/O 任务后才向 CPU 发出中断信号。

2) 通道分类

可对通道进行以下分类。

(1) 字节多路通道。通常都含有许多非分配型子通道，每一个子通道连接一台 I/O 设备。主通道采用时间片轮转法，轮流地为各个子通道服务。字节多路通道扫描通道的速率足够快，而连接到子通道的设备速率不太高时便不会丢失信息。

(2) 数组选择通道。由于字节多路通道不适于连接高速设备，所以引入数组选择通道。这种通道的传输速率高，可以连接多台高速设备，但由于该通道仅含有一个可分配型通道，因此在某一段时间内只能执行一个通道程序，为一台设备进行输入输出。

(3) 数组多路通道。结合了数组选择通道传输速率高和字节多路通道能使各个子通道分时并行操作的优点。该通道中含有多个非分配型子通道，因而该通道既具有很高的传输速率，又能获得令人满意的通道利用率，其数据传输是按数组方式进行的。

2. 直接存储访问方式

直接存储访问(Direct Memory Access，DMA)是指数据在主存和 I/O 设备间传送一个数据块的过程中，不需要 CPU 的任何干涉，只需要 CPU 在过程开始启动与过程结束时的处理，实际操作由 DMA 硬件直接执行完成，CPU 在此传送过程中可做别的事情。

3. 缓冲技术

缓冲技术可提高外设的利用率，尽可能使外设处于忙状态。缓冲技术可以分为硬件缓冲和软件缓冲技术。硬件缓冲是利用专门的硬件寄存器作为缓冲，软件缓冲是通过操作系统来管理的。引入缓冲的主要原因有以下几个方面。

(1) 缓和 CPU 与 I/O 设备间速度不匹配的矛盾。

(2) 减少对 CPU 的中断频率，放宽对中断响应时间的限制。

(3) 提高 CPU 和 I/O 设备之间的并行性。

在所有的 I/O 设备与处理机之间，都使用了缓冲区来交换数据。所以操作系统必须组织和管理好这些缓冲区。缓冲可以分为单缓冲、双缓冲、多缓冲和环形缓冲。

4. SPOOLing 技术

SPOOLing 技术实际上是用一类物理设备模拟另一类物理设备的技术，是使独占使用的

设备变成多台虚拟设备的一种技术，也是一种速度匹配技术。SPOOLing 系统是由“预输入程序”“缓输出程序”“井管理程序”以及输入输出井组成的。

SPOOLing 系统的工作过程是操作系统初启后激活 SPOOLing 预输入程序，使它处于捕获输入请求状态，一旦有输入请求消息，SPOOLing 输入程序立即得到执行，把装在输入设备上的作业输入到硬盘的输入井中，并填写好作业表以便在作业执行中要求输入信息时可以随时找到它们的存放位置。当作业需要输出数据时，可以先将数据送到输出井，当输出设备空闲时，由 SPOOLing 输出程序把硬盘上输出井的数据送到慢速的输出设备上。

SPOOLing 系统中拥有一张预输入表用来登记进入系统的所有作业的作业名、状态与输入表位置等信息。每个用户作业拥有一张预输入表用来登记该作业的各个文件的情况。输入井中的作业有以下 4 种状态。

- 提交状态：作业的信息正从输入设备上预输入。
- 后备状态：作业预输入结束但未被选中执行。
- 执行状态：作业已被选中并运行过程中，它可从输入井中读取数据信息，也可向输出井写信息。
- 完成状态：作业已经撤离，该作业的执行结果等待缓输出。

4.4.1.4　磁盘调度

磁盘是可被多个进程共享的设备。操作系统应采用一种适当的调度算法，以使各进程对磁盘的平均访问时间最小。磁盘调度分为移臂调度和旋转调度两类，并且是先进行移臂调度，然后再进行旋转调度。由于访问磁盘最耗时的是寻道时间，因此，磁盘调度的目标是使磁盘的平均寻道时间最少。

1. *磁盘驱动调度*

一般可采用以下 4 种磁盘调度算法。

(1) 先来先服务磁盘调度算法(FCFS)。这是最简单的磁盘调度算法。它根据进程请求访问磁盘的先后次序进行调度。优点是公平、简单，且每个进程的请求都能依次得到处理，不会出现某进程的请求长期得不到满足的情况。此算法由于未对寻道进行优化，因此平均寻道时间可能较长。

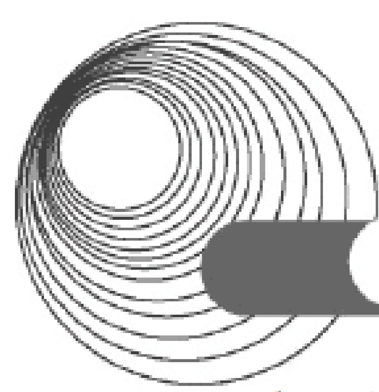
(2) 最短寻道时间优先磁盘调度算法(SSTF)。SSTF 算法要求访问的磁道与当前磁头所在的磁道距离最近，使得每次的寻道时间最短，但这种调度算法却不能保证平均寻道时间最短。

(3) 扫描算法(SCAN)。SCAN 算法也是一种寻道优化的算法，它克服了 SSTF 算法的缺点。既考虑访问磁道与磁头当前位置的距离，更优先考虑了当前的移动方向。这种算法磁头移动的规律颇似电梯的运行，故又常称为电梯调度算法。

(4) 单向扫描调度算法(CSCAN)。SCAN 存在这样的问题：当磁头刚从里向外移动过某一磁道时，恰有一进程请求访问此磁道，这时该进程必须等待磁头从里向外，然后再从外向里扫描完所有要访问的磁道后，才处理该进程的请求，致使该进程的请求被严重地推迟。为了减少这种延迟，CSCAN 算法规定了磁头做单向移动。

2. *旋转调度算法*

系统应该选择延迟时间最短的进程对磁盘的扇区进行访问。当有若干等待进程请求访



问磁盘上的信息时，旋转调度应考虑以下 3 种情况。

(1) 进程请求访问的是同一磁道上的不同编号的扇区。

(2) 进程请求访问的是不同磁道上的不同编号的扇区。

(3) 进程请求访问的是不同磁道上具有相同编号的扇区。

4.4.2 典型例题分析

例 1 假设磁盘每磁道有 18 个扇区，系统刚完成了 10 号柱面的操作，当前移动臂在 13 号柱面上，进程的请求序列如下表所示。若系统采用 SCAN(扫描)调度算法，则系统响应序列为 (26) ；若系统采用 CSCAN(单向扫描)调度算法，则系统响应序列为 (27) 。(2011 年下半年试题 26～27)

请求序列	柱 面 号	磁 头 号	扇 区 号
①	15	8	9
②	20	6	5
③	30	9	6
④	20	10	5
⑤	5	4	5
⑥	2	7	10
⑦	15	8	1
⑧	6	3	4
⑨	8	7	9
⑩	15	10	4

(26) A. ⑦⑩①②④③⑨⑧⑤⑥　　B. ①⑦⑩②③④⑥⑤⑧⑨

C. ⑦⑩①②④③⑥⑤⑧⑨　　D. ①⑦⑩②③④⑧⑨⑥⑤

(27) A. ⑦⑩①②④③⑨⑧⑤⑥　　B. ①⑦⑩②③④⑥⑤⑧⑨

C. ⑦⑩①②④③⑥⑤⑧⑨　　D. ①⑦⑩②③④⑧⑨⑥⑤

解析：SCAN 算法不仅要考虑欲访问的磁道与当前磁道的距离，更优先考虑磁头的当前移动方向。由题意知，当前磁头正在由里向外移动(因为从 10 号柱面移动到 13 号柱面)，所以下一个柱面应该是 15 号，题目中有 3 个柱面号为 15 的请求序列，选择扇区号最小的请求序列。如果和柱面号也相同，则选择扇区号小的，因此由里到外的系统响应序列为⑦→⑩→①→②→④→③。当移动臂位于 30 号柱面时，由于 30 号柱面是最外层的柱面，因此移动臂开始由外往里移动，后续的系统响应序列应为⑨→⑧→⑤→⑥。

CSCAN 在 SCAN 的基础上规定，磁头只能做单向移动，本题中只能由里往外移动，因此请求序列系统响应序列为⑦→⑩→①→②→④→③→⑥→⑤→⑧→⑨。

答案：(26) A　(27) C

例 2 假设磁盘块与缓冲区大小相同，每个盘块读入缓冲区的时间为 15μs，由缓冲区送至用户区的时间是 5μs，在用户区内系统对每块数据的处理时间为 1μs，若用户需要将大

小为 10 个磁盘块的 Doc1 文件逐块从磁盘读入缓冲区，并送至用户区进行处理，那么采用单缓冲区需要花费的时间为 (25) μs；采用双缓冲区需要花费的时间为 (26) μs。(2015 年下半年试题 25、26)

(25) A．150　　B．151　　C．156　　D．201

(26) A．150　　B．151　　C．156　　D．201

解析：单缓冲区情况下，当上一个磁盘块从缓冲区读入用户区完成时下一块磁盘块才能开始读入，也就是说，当最后一个磁盘块读入用户区完毕时所用的时间为(15+5) × 10=200，然后加上处理最后一个磁盘块的时间 1，得到 201。双缓冲区情况下，不存在等待磁盘块从缓冲区读入用户区的问题，因此把数据全部传输到缓冲区的时间为 15 × 10=150，再加上将双缓冲区的数据传输到用户区并处理完的时间 5+1=6，最后得到 156。

答案：(25) D　(26) C

例 3　在 Windows XP 操作系统中，用户利用“磁盘管理”程序可以对磁盘进行初始化、建卷，(23)。通常将 C:\Windows\myprogram.exe 文件设置成只读和隐藏属性，以便于控制用户对该文件的访问，这一级安全管理称为 (24) 安全管理。(2009 年上半年试题 23、24)

(23) A．但只能使用 FAT 文件系统格式化卷

B．但只能使用 FAT32 文件系统格式化卷

C．但只能使用 NTFS 文件系统格式化卷

D．可以选择使用 FAT、FAT32 或 NTFS 文件系统格式化卷

(24) A．文件级　　B．目录级　　C．用户级　　D．系统级

解析：对于固定磁盘来说，Microsoft Windows XP 支持 3 种文件系统，即 FAT16、FAT32 和 NTFS。

对于第 24 题，C:\Windows\myprogram.exe 其实就是文件 myprogram.exe 的绝对路径，所以把 C:\Windows\myprogram.exe 文件设置成只读和隐藏属性属于文件安全管理。

答案：(23) D　(24) A

例 4　在移臂调度算法中， (25) 算法可能会随时改变移动臂的运动方向。(2009 年上半年试题 25)

(25) A．电梯调度和先来先服务　　B．先来先服务和最短寻找时间优先

C．单向扫描和先来先服务　　D．电梯调度和最短寻找时间优先

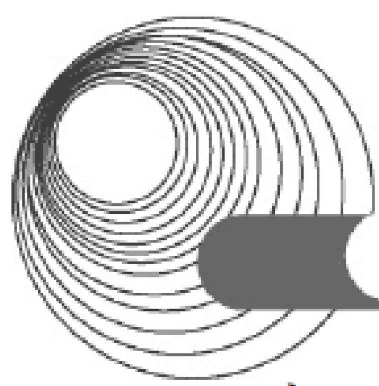
解析：常用的移臂调度算法有以下几种。

(1) 先来先服务算法。这个算法实际上不考虑访问者要求访问的物理位置，而只是考虑访问者提出访问请求的先后次序。故有可能随时改变移动臂的方向。

(2) 最短寻找时间优先算法。最短寻找时间优先调度算法总是从等待访问者中挑选寻找时间最短的那个请求先执行，而不管访问者到来的先后次序。故也有可能随时改变移动臂的方向。

(3) 电梯调度算法。电梯调度算法是从移动臂当前位置开始沿着臂的移动方向去选择离当前移动臂最近的那个访问者，如果沿臂的移动方向无请求访问时，就改变臂的移动方向再选择。

(4) 单向扫描调度算法。单向扫描调度算法的基本思想是，不考虑访问者等待的先后次序，总是从 0 号柱面开始向里道扫描，按照各自所要访问的柱面位置的次序去选择访问



者。在移动臂到达最后一个柱面后，立即快速返回到 0 号柱面，返回时不为任何的访问者等待服务。在返回到 0 号柱面后，再次进行扫描。

答案：B

例 5 假设磁盘块与缓冲区大小相同，每个盘块读入缓冲区的时间为 10μs，由缓冲区送至用户区的时间是 5μs，系统对每个磁盘块数据的处理时间为 2μs，若用户需要将大小为 10 个磁盘块的 Doc 文件逐块从磁盘读入缓冲区，并送至用户区进行处理，那么采用单缓冲区需要花费时间为 (25) μs；采用双缓冲区需要花费的时间为 (26) μs。(2014 年下半年试题 25、26)

(25)~(26) A．100　　B．107　　C．152　　D．170

解析：单缓冲区：(10+5)*10+2=152

双缓冲区：10*10+5+2=107

答案：(25)C　(26)B

4.4.3 同步练习

在操作系统中，通常临界资源不能采用_____分配算法。

A. 静态优先级　B. 动态优先级　C. 时间片轮转　D. 先来先服务

4.4.4 同步练习参考答案

C

4.5 文 件 管 理

4.5.1 考点辅导

4.5.1.1 文件与文件系统

1. 文件

文件是具有符号名的、在逻辑上具有完整意义的一组相关信息项的集合。

信息项是构成文件内容的基本单位，可以是一个字符，也可以是一个记录，记录可以等长，也可以不等长。一个文件包括文件体和文件说明。文件是一种抽象机制，它隐蔽了硬件和实现细节，提供了将信息保存在磁盘上而且便于以后读取的手段，使用户不必了解信息存储的方法、位置以及存储设备实际运作方式便可存取信息。在文件管理中的一个非常关键的问题在于文件的命名。不同的操作系统文件命名规则有所不同，即文件名字的格式和长度因系统而异。

2. 文件系统

文件管理系统就是操作系统中实现文件统一管理的一组软件和相关数据的集合。专门负责管理和存取文件信息的软件机构，简称文件系统。文件系统的功能包括按名存取、统一的用户接口、并发访问和控制、安全性控制、优化性能和差错恢复。

3. 文件类型

可以按不同的标准对文件进行分类。

(1) 按文件的性质和用途分类，可以分为系统文件、库文件和用户文件。

(2) 按信息保存期限分类，可以分为临时文件、档案文件和永久文件。

(3) 按文件的保护方式分类，可以分为只读文件、读写文件、可执行文件和不保护文件。

(4) UNIX 系统将文件分为普通文件、目录文件和特殊文件。

(5) 目前常用的文件系统类型有 FAT、VFAT、NTFS、EXT2、HPFS 等。

文件分类的目的是对不同文件进行管理，提高系统效率和用户界面的友好性。

4.5.1.2　文件的结构和组织

文件的结构是指文件的组织形式，从用户观点所看到的文件组织形式，称为文件的逻辑结构；从实现观点考察文件在辅助存储器上的存放方式，常称为文件的物理结构。

1. 文件的逻辑结构

文件的逻辑结构可以分为两类：一类是有结构的记录式文件，它是由一个以上的记录构成的文件；另一类是无结构的流式文件，它是由一串顺序字符流构成的文件。

(1) 有结构的记录式文件。记录文件根据长度可分为定长和不定长两种。

(2) 无结构的流式文件。无结构的流式文件通常采用顺序访问方式，并且每次读写访问可以指定任意数据长度，其长度以字节为单位。

2. 文件的物理结构

文件的物理结构是指文件的内部组织形式，也就是文件在物理存储设备上的存放方法。常用的文件物理结构有以下 3 种。

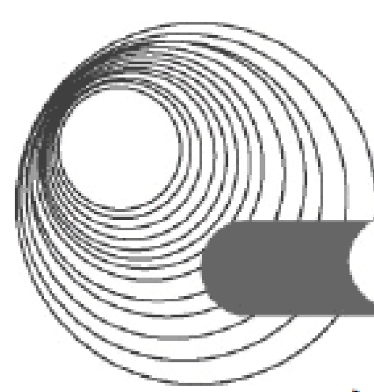
(1) 连续结构。连续结构也称为顺序结构。这是一种最简单的物理结构，它把逻辑上连续的文件信息依次存放在连续编号的物理块中。只要知道文件在存储设备上的起始地址(首块号)和文件长度(总块数)，就能很快地进行存取。这种结构的缺点是不便于记录的增加或删除操作。

(2) 链接结构。链接结构也叫串联结构。它是将逻辑上连续的文件信息存放在不连续的物理块中，每个物理块设有一个指针指向其下一个物理块。只要指明文件的第一个物理块号，就可以利用链指针检索整个文件。

(3) 索引结构。采用索引结构将逻辑上连续的文件信息存放在不连续的物理块中，系统为每个文件建立一张索引表。索引表记录了文件信息所在的逻辑块号对应的物理块号，并将索引表的起始地址放在文件对应的文件目录项中。

多个物理块的索引表分为两种组织方式，即链接文件和多重索引方式。

UNIX 文件系统采用的是 3 级索引结构，文件系统中 inode 是基本的构件，它表示文件系统树形结构的节点(注：树形结构也称为树型结构或树状结构)。UNIX 有直接、一级间接、



二级间接、三级间接 4 种寻址方式。

4.5.1.3 文件目录

系统为每个文件设置一个描述性数据结构——文件控制块(File Control Block，FCB)，文件目录就是文件控制块的有序集合。

1. 文件控制块

FCB 是系统为管理文件而设置的一个数据结构。FCB 是文件存在的标志，它记录了系统管理文件所需要的全部信息。FCB 通常应包括 3 类信息。

(1) 基本信息类：文件名、文件的物理位置、文件长度、文件块数等。

(2) 存取控制信息类：文件的存取权限。

(3) 使用信息类：文件的建立日期、最后一次修改的日期、最后一次访问的日期，当前使用的信息和目录文件。

2. 目录结构

文件目录结构的组织方式直接影响到文件的存取速度，关系到文件共享性和安全性。常见的目录结构有 3 种，即一级目录结构、二级目录结构和多级目录结构。目前大多数操作系统(如 UNIX、DOS 等)都采用多级目录结构，又称为树形目录结构。

1) 一级目录结构

一级目录的整个目录组织是一个线性结构，在整个系统中只需建立一张目录表，系统为每个文件分配一个目录项(文件控制块)。一级目录结构简单，但缺点是查找速度慢，不允许重名和不便于实现文件共享等，因此它主要用在单用户环境中。

2) 二级目录结构

二级目录结构是由主文件目录(Master File Directory，MFD)和用户目录(User File Directory，UFD)组成的。在主文件目录中，每个用户文件目录都占有一个目录项，其目录项中包括用户名和指向该用户目录文件的指针。用户目录由用户所有文件的目录项组成。

二级目录的优点是提高了检索目录的速度，较好地解决了重名问题。采用二级目录结构也存在一些问题。该结构虽然能有效地将多个用户隔离开(这种隔离在各个用户之间完全无关时是一个优点)，但当多个用户之间要相互合作去共同完成一个大任务时，且一用户又要去访问其他用户的文件时，这种隔离便成为一个缺点，因为这种隔离使诸用户之间不便于共享文件。

3) 多级目录结构

为了解决以上问题，在多道程序设计系统中常采用多级目录结构，这种目录结构就像一棵倒置的有根树，所以也称为树形目录结构。从树根向下，每一个节点是一个目录，叶节点是文件。MS-DOS 和 UNIX 等操作系统均采用多级目录结构。

采用多级目录结构的文件系统中，用户要访问一个文件，必须指出文件所在的路径名。路径名包含从根目录开始到该文件的通路上所有各级目录名。各级目录名之间、目录名与文件名之间需要用分隔符隔开。例如，在 DOS 中分隔符为“\”，在 UNIX 中分隔符为“/”。绝对路径名是指从根目录开始的完整文件名，即由从根目录开始的所有目录名以及文件名构成的。

4.5.1.4　存取方法和存取空间的管理

1. 文件的存取方法

文件的存取方法是指读写文件存储器上的一个物理块的方法。通常有顺序存取、随机存取和按键存取等。

1)　顺序存取

顺序存取就是按从前到后的次序依次访问文件的各个信息项。对于记录式文件，是按物理记录的排列顺序来存取的。

2)　随机存取

随机存取又称为直接存取，即允许用户随意存取文件的任意一个物理记录。

3)　按键存取

按键存取是直接存取法的一种，它不是根据记录的编号或地址来存取文件中的记录，而是根据文件中各记录的某个数据项内容来存取记录的，这种数据项称为“键”。因此，将这种存取法称为按键存取。

2. 文件存储空间的管理

外存空间管理的数据结构通常称为磁盘分配表。常用的空间管理方法有空闲区表、位示图、空闲块链和成组链接法 4 种。

1)　空闲区表

将外存空间上一个连续未分配区域称为空闲区。操作系统为磁盘外存上所有空闲区建立一张空闲表，每个表项对应一个空闲区，空闲表中包含序号、空闲区的第一块号、空闲块的块数等信息。它适用于连续文件结构。

2)　位示图

在外存上建立一张位示图，记录文件存储器的使用情况。每一位对应文件存储器上的一个物理块，取值 0 和 1 分别表示空闲和占用。这种方法的主要特点是位示图的大小由磁盘空间的大小(物理块总数)决定，位示图的描述能力强，适合各种物理结构。

3)　空闲块链

每个空闲物理块中有指向下一个空闲物理块的指针，所有空闲物理块构成一个链表，链表的头指针放在文件存储器的特定位置上(如管理块中)。

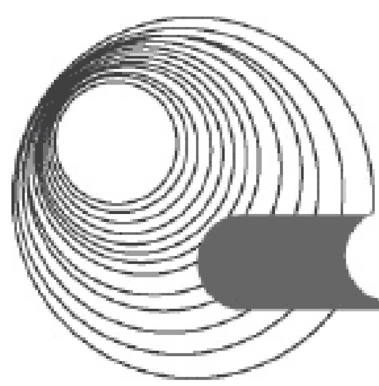
4)　成组链接法

在 UNIX 系统中，将空闲块分成若干组，每 100 个空闲块为一组，每组的第一个空闲块登记了下一组空闲块的物理盘块号和空闲块总数，假如一个组的第一个空闲块号等于 0 的话，就意味着该组是最后一组，即无下一组空闲块。

4.5.1.5　文件的使用

操作系统在操作级(命令级)和编程级(系统调用和函数)向用户提供文件的服务。操作系统在操作级向用户提供的命令有目录管理类命令、文件操作类命令(如复制、删除和修改)、文件管理类命令(如设置文件权限)等。操作系统在编程级向用户提供的系统调用主要有以下 6 种。

- 创建文件：如 create(文件名，参数表)。



- 删除文件：如 delete(文件名)。
- 打开文件：如 open(文件名，参数表)。
- 关闭文件：如 close(文件名)。
- 读文件：如 read(文件名，参数表)。
- 写文件：如 write(文件名，参数表)。

4.5.1.6 文件的共享和保护

1. 文件的共享

文件共享是指不同用户进程使用同一文件。文件共享有多种形式，采用文件名和文件说明分离的目录结构有利于实现文件共享。常见的文件链接有硬链接和符号链接两种。

1) 硬链接

文件的硬链接是指两个文件目录表目指向同一个索引节点的链接，该链接也称为基于索引节点的链接。文件硬链接不利于文件主删除它拥有的文件，因为文件主要删除它拥有的共享文件，必须首先删除(关闭)所有的硬链接；否则就会造成共享该文件的用户目录表目指针悬空。

2) 符号链接

符号链接是指建立的新的文件或目录与原来文件或目录的路径名映射。当访问一个符号链接时，系统通过该映射找到源文件的路径，并对其进行访问。符号链接的缺点是：其他用户读取符号链接的共享文件比读取硬链接的共享文件需要增加读盘操作的次数。因为其他用户去读符号链接的共享文件时，系统中根据给定的文件路径名，逐个去查找目录，通过多次读盘操作才能找到该文件的索引节点，而用硬链接的共享文件的目录文件表目中已包括了共享文件的索引节点号。

2. 文件的保护

文件系统对文件的保护常采用存取控制方式进行。存取控制就是不同的用户对文件的访问有不同的权限，以防止文件被未经文件主同意的用户访问。

1) 存取控制矩阵

理论上，存取控制可用存取控制矩阵方法，它是一个二维矩阵，一维列出计算机的全部用户，另一维列出系统中的全部文件。存取控制矩阵在概念上是简单清楚的，但实际上却有困难。当一个系统用户数和文件数很大时，二维矩阵要占很大的存储空间，验证过程也将耗费许多系统时间。

2) 存取控制表

存取控制表是按用户对文件访问权限的差别对用户进行分类，由于某一文件往往只与少数几个用户有关，所以这种分类方法可使存取控制表大为简化。UNIX 系统就是使用这种存取控制表方法。它把用户分成三类，包括文件主、同组用户和其他用户，每类用户的存取权限为可读、可写、可执行及其组合。

3) 用户权限表

用户权限表是以用户或用户组为单位将用户可存取的文件集中起来存入表中，表中每个表目表示该用户对相应文件的存取权限，这相当于存取控制矩阵一行的简化。

4)　密码

在创建文件时，由用户提供一个密码，在文件存入磁盘时用该密码对文件内容加密。进行读取操作时，要对文件进行解密，只有知道密码的用户才能读取文件。

4.5.1.7　系统的安全与可靠性

1. 系统的安全

系统的安全涉及两类问题：一类涉及技术、管理、法律、道德和政治等问题；另一类涉及操作系统的安全机制。

一般从 4 个级别上对文件进行安全性管理，即系统级、用户级、目录级和文件级。

(1) 系统级安全管理的主要任务是不允许未经授权的用户进入系统，从而也防止了他人非法使用系统中的各类资源(包括文件)。系统级管理的主要措施有注册与登录。

(2) 用户级安全管理是对通过分类的所有用户和对指定的用户分配访问权。不同的用户通过对不同文件设置不同的存取权限来实现安全管理。有的系统将用户分为超级用户、系统操作员和一般用户。

(3) 目录级安全管理是为了保护系统中各种目录而设计的，与用户权限无关。为保证目录的安全，规定只有系统核心才具有写目录的权利。

(4) 文件级安全管理是通过系统管理员或文件主对文件属性的设置来控制用户对文件的访问。通常可设置的属性有只执行、隐含、只读、读写、共享、系统。

2. 文件系统的可靠性

文件系统的可靠性是指系统抵抗和预防各种物理破坏和人为破坏的能力。如果文件系统被破坏了，在很多情况下是无法恢复的。

1)　转储和恢复

文件系统中无论是硬件还是软件都会发生损坏和错误，为了使文件系统万无一失，应当采用相应的措施。最简单和常用的措施是通过转储操作，制作文件或文件系统的多个副本。这样一旦系统出现故障，利用转储的数据可使系统恢复成为可能。常用的转储方法有静态转储和动态转储、海量转储和增量转储。

2)　日志文件

在计算机系统工作的过程中，操作系统把用户对文件的插入、删除和修改的操作写入日志文件。一旦发生故障，操作系统的恢复子系统就利用日志文件来进行系统故障恢复，并可协助后备副本进行介质故障恢复。

3)　文件系统的一致性

影响文件系统可靠性因素之一是文件系统的一致性问题。很多文件系统是先读取磁盘块到主存，在主存进行修改，修改完毕再写回磁盘。但如果读取某磁盘块，修改后再将信息写回磁盘前系统崩溃，则文件系统就可能会出现不一致性状态。如果这些未被写回的磁盘块是索引节点块、目录块或空闲块，那么后果是不堪设想的。通常解决方案是采用文件系统的一致性检查，包括块的一致性检查和文件的一致性检查。

4.5.2 典型例题分析

例 1 设文件索引节点中有 8 个地址项，每个地址项大小为 4B，其中 5 个地址项为直接地址索引，两个地址项为一级间接索引，一个地址项为二级间接索引。磁盘索引块和磁盘数据块大小为 1KB。若要访问文件的逻辑块号分别为 8 和 518，则系统应分别采用 (27) ，而且可以表示的单个文件最大长度为 (28) KB。(2012 年下半年试题 27、28)

(27) A．直接地址索引和一级间接地址索引
B．直接地址索引和二级间接地址索引
C．一级间接地址索引和二级间接地址索引
D．一级间接地址索引和一级间接地址索引

(28) A．517　　B．1029　　C．16513　　D．66053

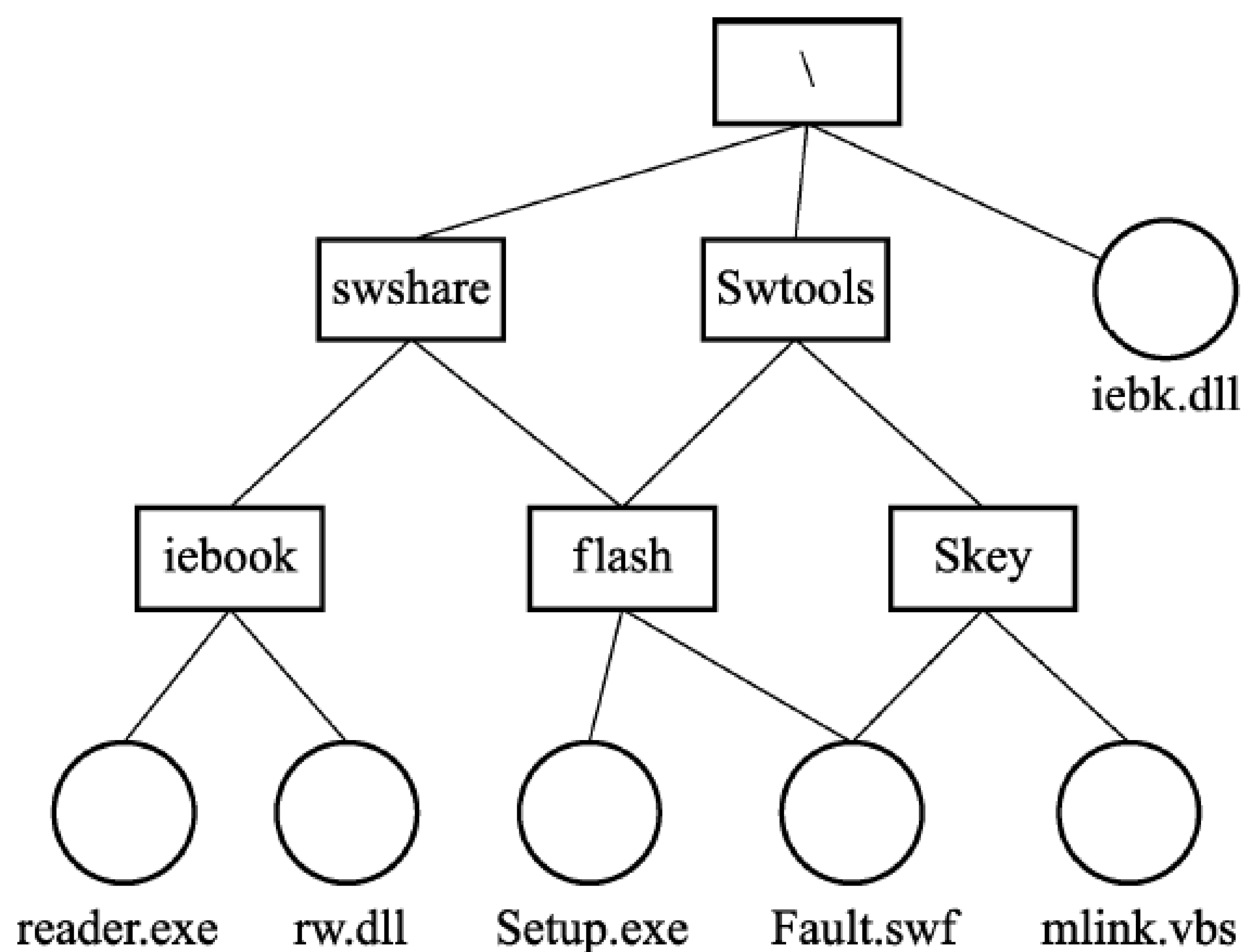
解析：若逻辑块号<5，则为直接块，使用直接地址索引；若 5≤逻辑块号<261(2^8+5)，则采用一级间接地址索引；若 261≤逻辑块号<65797(2^{16}+261)，则采用二级间接地址索引。因此若要访问文件的逻辑块号分别为 8 和 518，则系统应分别采用一级间接地址索引和二级间接地址索引。

磁盘索引块的大小为 1KB，每个盘块号为 4B，则一个磁盘块可以保存的盘块号的个数为 1K/4=256。

因每个磁盘索引块和磁盘数据块大小均为 1KB，所以 5 个直接地址索引指向的数据块大小为 5KB，两个一级间接索引共包括 2×256 个直接地址索引，其指向的数据块大小为 2×256×1KB=512KB，一个二级间接索引所包含的直接地址索引为 256×256×1KB=65536KB。8 个地址项所指向的数据块总大小即单个文件最大长度为(65536+512+5)KB，即 66053KB。

答案：(27) C　　(28) D

例 2 若某文件系统的目录结构如下图所示，假设用户要访问文件 fault.swf，且当前工作目录为 swshare，则该文件的全文件名为(27)，相对路径和绝对路径分别为(28)。(2014 年上半年试题 27、28)



(27) A．fault.swf　　B．flash\fault.swf
C．swshare\flash\fault.swf　　D．\swshare\flash\fault.swf

(28) A．swshare\flash\和\flash\　　B．flash\和\swshare\flash\
C．\swshare\flash\和 flash\　　D．\flash\和\swshare\flash\

解析：全文件名应该从根目录\swshare\flash\fault.swf。相对路径是从当前路径开始的路径，fault.swf 在当前工作目录 swshare 下的 flash 文件夹中，因此相对路径为 flash\。绝对路径是指从根目录开始的路径，即\swshare\flash\。

答案：(27)D　(28)B

例 3　某文件管理系统在磁盘上建立了位示图(bitmap)，记录磁盘的使用情况。若系统的字长为 32bit，磁盘上的物理块依次编号为 0、1、2、…，那么 4096 号物理块的使用情况在位示图中的第<u>(23)</u>个字中描述；若磁盘的容量为 200GB，物理块的大小为 1MB，那么位示图的大小为<u>(24)</u>个字。(2011 年上半年试题 23、24)

(23) A．129　B．257　C．513　D．1025
(24) A．600　B．1200　C．3200　D．6400

解析：系统的字长为 32bit，在位示图中的第一个字对应文件存储器上的 0、1、2、3、…31 号物理块，以此类推。4096/32=128，4096 号物理块是第 129 个字对应的第一个物理块。磁盘的容量为 200GB，物理块的大小为 1MB，则磁盘共 200×1024 个物理块，一个字对应 32 个物理块，位示图的大小为 200×1024/32 =6400 个字。

答案：(23)A　(24)D

例 4　正常情况下，操作系统对保存有大量有用数据的硬盘进行<u>(6)</u>操作时，不会清除有用数据。(2010 年下半年试题 6)

(6) A．磁盘分区和格式化　　B．磁盘格式化和碎片整理
C．磁盘清理和碎片整理　　D．磁盘分区和磁盘清理

解析：计算机中存放信息的主要存储设备就是硬盘，但是硬盘不能直接使用，必须对硬盘进行分割，分割成的一块一块的硬盘区域就是磁盘分区。磁盘分区后，必须经过格式化才能正式使用。磁盘格式化是在物理驱动器(磁盘)的所有数据区上写零的操作过程。磁盘清理是清除没用的文件，以节省磁盘空间。磁盘碎片整理是通过系统软件或者专业的磁盘碎片整理软件对计算机磁盘在长期使用过程中产生的碎片和凌乱文件重新整理，释放出更多的磁盘空间，可提高计算机的整体性能和运行速度。

答案：C

例 5　某文件系统采用多级索引结构，若磁盘块的大小为 512B，每个块号需占 3B，那么根索引采用一级索引时的文件最大长度为<u>(27)</u>KB；采用二级索引时的文件最大长度为<u>(28)</u>KB。(2010 年下半年试题 27、28)

(27) A．85　B．170　C．512　D．1024
(28) A．512　B．1024　C．14450　D．28900

解析：多级索引结构采用了间接索引方式，第一级索引表的表目指出下一级索引表的位置(物理块号)，下一级索引表的表目指出再下一级索引表的位置，这样间接几级，最末一级索引表的表目则指向相应记录所在的物理块号。一级索引可存放 512/3=170 个物理块块号，每个块大小 512B，文件最大长度 170×512/1024=85KB，选择 A。二级索引可存放 170×170 个物理块块号，每个块大小 512B，文件最大长度 170×170×512/1024=14450(KB)，选择 C。

答案：(27) A　(28) C

例 6　若系统正在将<u>(27)</u>文件修改的结果写回磁盘时系统发生崩溃，则对系统的影响

相对较大。(2009 年下半年试题 27)

(27) A．空闲块　　B．目录　　C．用户数据　　D．用户程序

解析：磁盘有以下几种写方式。

一般写方式：这是真正把缓冲区中的数据写到磁盘上，且进程须等待写操作完成，由 bwrite 过程完成。

异步写方式：进程无须等待写操作完成便可返回，异步写过程是 bawrite。

延迟写方式：该方式并不真正启动磁盘，而只是在缓冲首部设置延迟写标志，然后便释放该缓冲区，并将之链入空闲链表的末尾。以后，当有进程申请到该缓冲区时，才将其内容写入磁盘。引入延迟写的目的是为了减少不必要的磁盘 I/O，因为只要没有进程申请到此缓冲区，其中的数据便不会被写入磁盘，倘若再有进程需要访问其中的数据时，便可直接从空闲链表中摘下该缓冲区，而不必从磁盘读入。延迟写方式由过程 bdwrite 完成。

当文件处于“未打开”状态时，文件需占用以下 3 种资源。

(1) 一个目录项。

(2) 一个磁盘索引节点项。

(3) 若干个盘块。

当文件被引用或“打开”时，须再增加以下 3 种资源。

(1) 一个内存索引节点项。它驻留在内存中。

(2) 文件表中的一个登记项。

(3) 用户文件描述符表中的一个登记项。

由于对文件的读写管理，必须涉及上述各种资源，对文件的读写管理，又在很大程度上依赖于对这些资源的管理，故可从资源管理观点上来介绍文件系统。这样，对文件的管理就必然包括：①对索引节点的管理；②对空闲盘块的管理；③对目录文件的管理；④对文件表和描述符表的管理；⑤对文件的使用。

因此如果目录文件在写回磁盘时发生异常，对系统的影响是很大的。而空闲块、用户数据和程序异常并不影响系统的工作，因此不会有较大的影响。

答案：B

例 7　某文件系统采用链式存储管理方案，磁盘块的大小为 1024B。文件 Myfile.doc 由 5 个逻辑记录组成，每个逻辑记录的大小与磁盘块的大小相等，并依次存放在 121、75、86、65 和 114 号磁盘块上。若需要存取文件的第 5120 逻辑字节处的信息，应该访问(28)号磁盘块。(2009 年上半年试题 28)

(28) A．75　　B．85　　C．65　　D．114

解析：每个逻辑记录的大小与磁盘块的大小相等(1024B)，当需要存取文件的第 5120 逻辑字节处的信息时，由计算可得 $5120=1024\times5$，即该处信息在第五个逻辑记录上，对应的是 114 号磁盘块。

答案：D

4.5.3　同步练习

1.　某文件管理系统为了记录磁盘的使用情况，在磁盘上建立了位示图。若系统中字长为 16 位，磁盘上的物理块依次编号为 0、1、2、…，那么 8192 号物理块的使用情况在位

示图中的第_____个字中描述。

A．256　　B．257　　C．512　　D．513

2．若文件系统允许不同用户的文件可以具有相同的文件名，则操作系统应采用_____来实现。

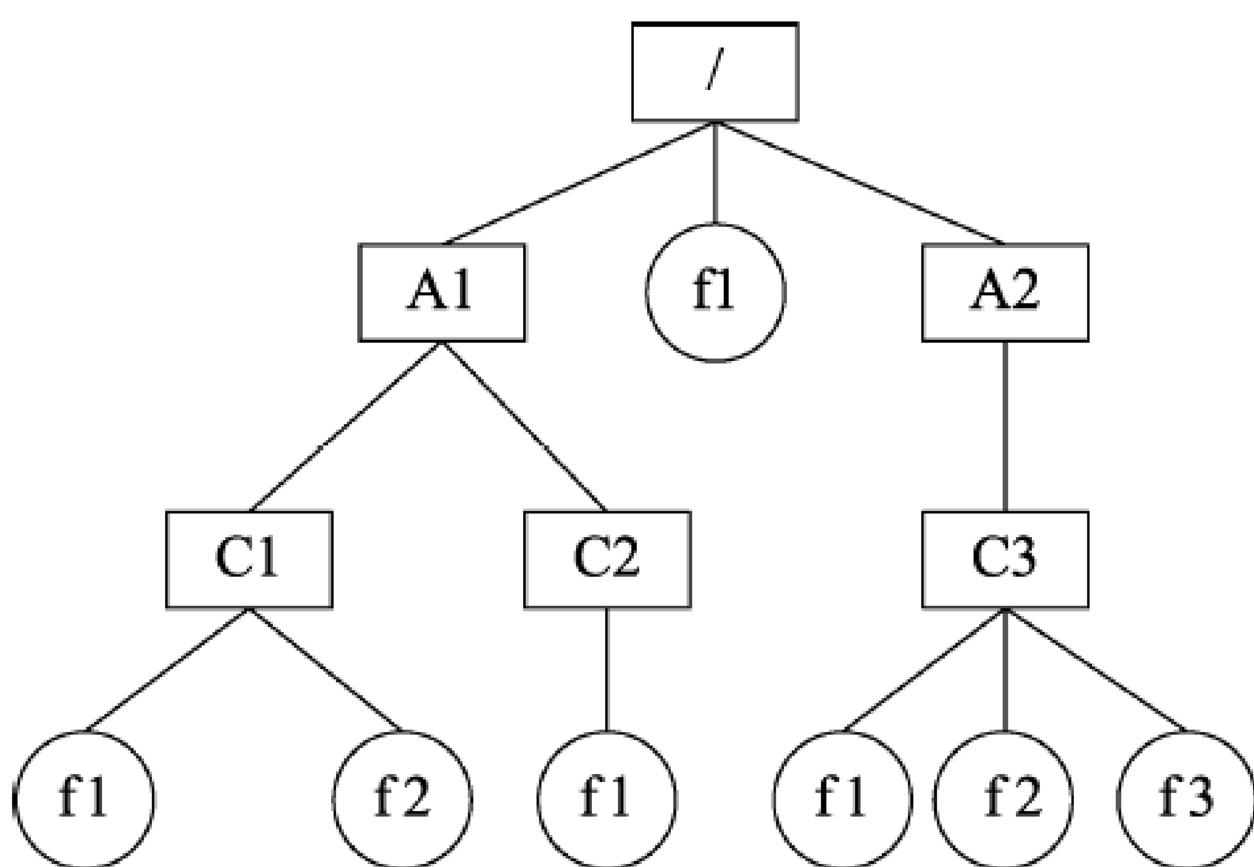
A．索引表　　B．索引文件　　C．指针　　D．多级目录

3．在下图所示的树形文件系统中，方框表示目录，圆圈表示文件，“/”表示路径中的分隔符，“/”在路径之首时表示根目录。图中，(1)。假设当前目录是 A2，若进程 A 以下面两种方式打开文件 f2：

方式①　fd1=open("(2)/f2"，o_RDONLY);

方式②　fd1=open("/A2/C3/f2"，o_RDONLY);

那么，采用方式①的工作效率比方式②的工作效率高。



(1) A．根目录中文件 f 1 与子目录 C1、C2 和 C3 中文件 f 1 一定相同

B．子目录 C1 中文件 f 2 与子目录 C3 中文件 f 2 一定相同

C．子目录 C1 中文件 f 2 与子目录 C3 中文件 f 2 一定不同

D．子目录 C1 中文件 f 2 与子目录 C3 中文件 f 2 可能相同也可能不相同

(2) A．/A2/C3　　B．A2/C3　　C．C3　　D．f2

4.5.4　同步练习参考答案

1．D

2．D

3．(1) D　　(2) C

4.6　作 业 管 理

4.6.1　考点辅导

4.6.1.1　作业管理和作业控制

作业是系统为完成一个用户的计算任务(或一次事务处理)所做的工作总和。操作系统中

用来控制作业的进入、执行和撤销的一组程序称为作业管理程序，这些控制功能通过把作业步细化和执行进程来实现。

1. 作业控制

用户作业可以采用脱机和联机两种控制方式控制作业的运行。在脱机控制方式中，作业运行的过程是无需人工干预的，因此用户必须将自己的意图用作业控制语言(JCL)编写成作业说明书连同作业一起提交给计算机系统。在联机控制方式中，操作系统向用户提供了一组联机命令，用户可以通过终端输入命令，将自己的意图告诉计算机，以控制作业的运行过程，因此整个作业的运行过程需要人工干预。

作业由程序、数据和作业说明书三部分组成。作业说明书包括作业基本情况、作业控制的描述、作业资源要求的描述。它体现了用户的控制意图。其中，作业基本情况包括用户名、作业名、编程语言、最大处理时间等；作业控制描述包括作业控制方式、作业步的操作顺序、作业执行出错处理；作业资源要求的描述包括处理时间、优先级、主存空间、外设类型和数量、实用程序要求等。

2. 作业状态及其转换

作业的状态分为 4 种，即提交、后备、执行和完成。

(1) 提交。作业提交给计算机中心，通过输入设备送入计算机系统的过程状态称为提交状态。

(2) 后备。作业通过 SPOOLing 系统输入到计算机系统的后备存储器(磁盘)中，随时等待作业调度程序调度时的状态。

(3) 执行。一旦作业被作业调度程序选中，为其分配了必要的资源，并为其建立了相应的进程后，该作业便进入了执行状态。

(4) 完成。当作业正常结束或异常终止时，作业进入完成状态。此时由作业调度程序对该作业进行善后处理。如撤销作业的作业控制块，收回作业所占的系统资源，将作业的执行结果形成输出文件放到输出井中，由 SPOOLing 系统控制输出。

3. 作业控制块和作业后备队列

作业控制块(JCB)是记录与该作业有关的各种信息的登记表。作业控制块是作业存在的唯一标志，包括用户名、作业名、状态标志等信息。

由于在输入井中有较多的后备作业，为了便于作业调度程序调度，通常将作业控制块排成一个或多个队列，这些队列称为作业后备队列，即作业后备队列是由若干个作业控制块组成的。

4.6.1.2 作业调度

1. 作业调度算法

常见的作业调度算法如下。

(1) 先来先服务(FCFS)。按作业到达先后进行调度，即启动等待时间最长的作业。

(2) 短作业优先(SJF)。以要求运行时间长短进行调度，即启动要求运行时间最短的作业。

(3) 响应比高优先(HRN)。定义响应比，即 HRN=作业响应时间/作业执行时间，其中作业响应时间是作业进入系统后的等待时间与作业的执行时间之和。

(4) 优先级调度算法。可由用户指定作业优先级，根据作业的优先级别，优先级高者先调度。

(5) 均衡调度算法。根据系统的运行情况和作业本身的特性对作业进行分类。作业调度程序轮流地从这些不同类别的作业中挑选执行。这种算法力求均衡地使用系统的各种资源，既注意发挥效率，又使用户满意。

2. 作业调度算法性能的衡量指标

在一个以批量处理为主的系统中，通常用平均周转时间或平均带权周转时间来衡量调度性能的优劣。假设作业 J_i(i=1，2，…，n)的提交时间为 t_{si}，执行时间为 t_{ri}，作业完成时间为 t_{oi}，则作业 J_i 的周转时间 T_i 和周转系数 W_i 分别定义为

$$T_i=t_{oi}-t_{si} \quad i=1，2，…，n$$

$$W_i=T_i/t_{ri} \quad i=1，2，…，n$$

n 个作业的平均周转时间 T 和平均带权周转时间 W 分别定义为

$$T=\frac{1}{n}\sum_{i=1}^{n}T_i$$

$$W=\frac{1}{n}\sum_{i=1}^{n}W_i$$

从用户的角度来说，总是希望自己的作业在提交后能立即执行，这意味着当等待时间为零时作业的周转时间最短，即 $T_i=t_{ri}$。但是作业的执行时间 t_{ri} 并不能直观地衡量出系统的性能，而带权周转时间 W_i 却能直观地反映系统的调度性能。从整个系统的角度来说，不可能满足每个用户的这种要求，而只能是系统的平均周转时间或平均带权周转时间最小。

4.6.1.3　用户界面

用户界面是计算机中实现用户与计算机通信的软件和硬件部分的总称。用户界面也称为用户接口或人机界面。

1. 控制面板式用户界面

计算机发展早期，用户通过控制台开关、板键或穿孔纸带向计算机送入命令或数据，而计算机通过指示灯及打印机输出运行情况或结果。这种界面的特点是人去适应计算机，现在看来是十分笨拙的。

2. 字符用户界面

字符用户界面是基于字符型的。用户通过键盘或其他输入设备输入字符，由显示器或打印机输出字符。字符用户界面的优点是功能强、灵活性好、屏幕开销少；缺点是操作步骤繁琐，学会操作较费时。

3. 图形用户界面

随着文字、图形、声音、图像等多媒体技术的出现，各种图形用户界面应运而生，用户既可使用传统的字符，也可以使用图形、图像和声音同计算机进行交互，操作更为自然、

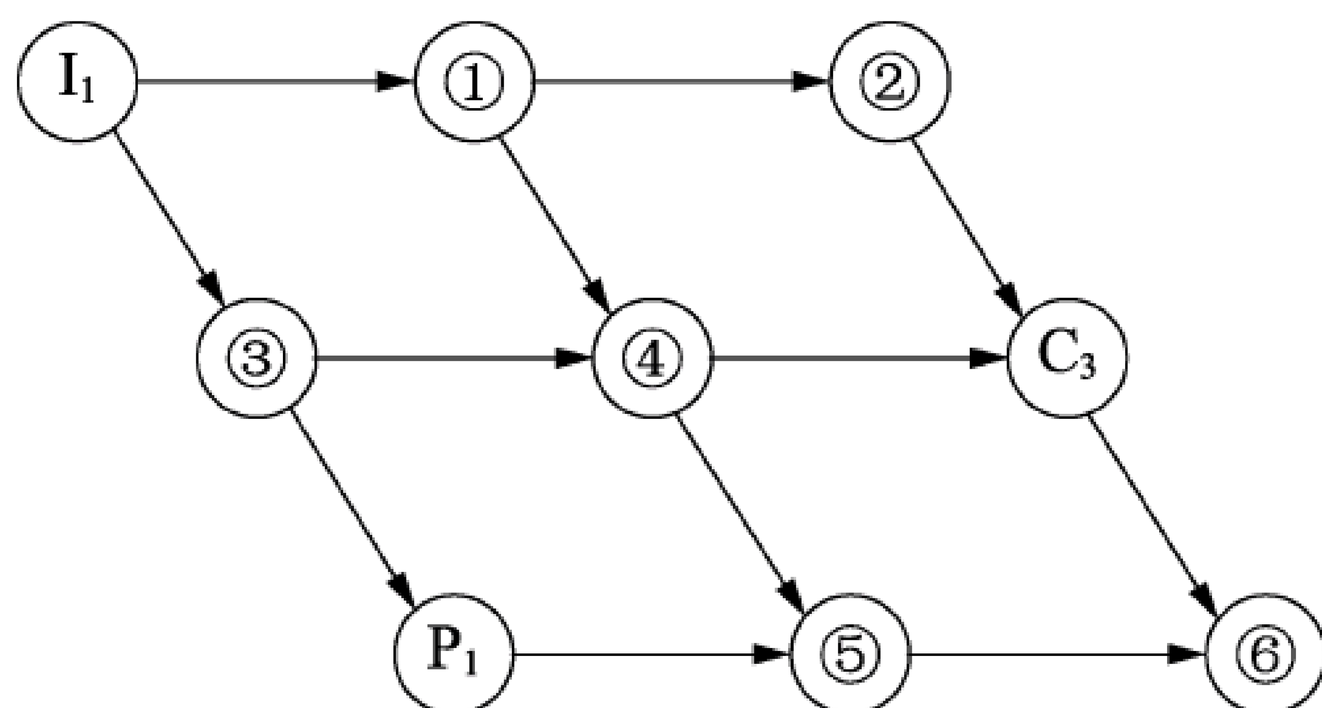
更加方便，多媒体技术进一步推广、发展与完善。现代界面的关键技术是超文本。超文本的“超”体现在它不仅包括文本，还包括图像、音频、视频等多媒体信息，即将文本的概念扩充到超文本，超文本的最大特点是具有指向性。

4. 新一代用户界面

多媒体、多通道及智能化是新一代用户界面的技术支持。新的、更加自然的交互技术，将为用户提供更方便的输入技术。计算机将通过多种感知通道来理解用户的意图，实现用户的要求；计算机不仅以二维屏幕向用户输出，而且能以真实感的计算机仿真环境向用户提供真实的体验。

4.6.2 典型例题分析

例 1 某计算机系统中有一个 CPU、一台输入设备和一台输出设备，假设系统中有 3 个作业 T_1、T_2 和 T_3，系统采用优先级调度，且 T_1 的优先级>T_2 的优先级>T_3 的优先级。若每个作业具有 3 个程序段：输入 I_i、计算 C_i 和输出 P_i(i=1,2,3)，执行顺序为 I_i、C_i、P_i，则这 3 个作业各程序段并发执行的前趋图如下图所示。图中①、②分别为 (24) ，③、④分别为 (25) ，⑤、⑥分别为 (26) 。(2012 年下半年试题 24～26)



(24) A. I_2、C_2　　B. I_2、I_3　　C. C_1、P_3　　D. C_2、P_2

(25) A. C_1、C_2　　B. I_2、C_1　　C. I_3、P_3　　D. C_1、P_2

(26) A. I_3、C_2　　B. I_2、C_1　　C. P_2、P_3　　D. C_1、P_2

解析：T_1 的优先级最高，则首先从作业 T_1 开始执行。当 T_1 的输入完成后，便可进行计算；同时输入设备空闲，可以执行作业 T_2 的输入操作，由此可以判断①和③中一个是 I_2 一个是 C_1。很显然，P_1 的前驱是 C_1，因为只有 C_1 执行完后，输入设备才会空闲，输入设备空闲后，才能执行 P_1。由此得知③为 C_1，则①为 I_2。

而以 C_1 为前驱的，一是 P_1，二是 C_2，故④为 C_2。T_1 的计算完成后，便可进行输出操作，同时计算设备空闲，可以进行 T_2 的计算操作，很显然，②为 I_3。

以此类推，得知⑤、⑥为 P_2、P_3。

答案：(24) B　(25) A　(26) C

例 2 嵌入式系统初始化过程主要有 3 个环节，按照自底向上、从硬件到软件的次序依次为(27)。系统级初始化主要任务是(28)。(2015 年上半年试题 27、28)

(27) A. 片级初始化→系统级初始化→板级初始化

B. 片级初始化→板级初始化→系统级初始化

C．系统级初始化→板级初始化→片级初始化
D．系统级初始化→片级初始化→板级初始化

(28) A．完成嵌入式微处理器的初始化
B．完成嵌入式微处理器以外的其他硬件设备的初始化
C．以软件初始化为主，主要进行操作系统的初始化
D．设置嵌入式微处理器的核心寄存器和控制寄存器工作状态

解析：系统初始化过程可以分为 3 个主要环节，按照自底向上、从硬件到软件的次序依次为片级初始化、板级初始化和系统级初始化。

片级初始化：完成嵌入式微处理器的初始化，包括设置嵌入式微处理器的核心寄存器和控制寄存器、嵌入式微处理器核心工作模式和嵌入式微处理器的局部总线模式等。

板级初始化：完成嵌入式微处理器以外的其他硬件设备的初始化。另外，还需设置某些软件的数据结构和参数，为随后的系统级初始化和应用程序的运行建立硬件和软件环境。

系统初始化：该初始化过程以软件初始化为主，主要进行操作系统的初始化。板级支撑包(BSP)将对嵌入式微处理器的控制权转交给嵌入式操作系统，由操作系统完成余下的初始化操作，包含加载和初始化与硬件无关的设备驱动程序，建立系统内存区，加载并初始化其他系统软件模块，如网络系统、文件系统等。最后，操作系统创建应用程序环境，并将控制权交给应用程序的入口。

答案：(27) B　　(28) C

4.6.3　同步练习

在有一台处理机 CPU 和两台输入输出设备 IO1 和 IO2，且能够实现抢先式多任务并行工作的多道程序内，投入运行优先级由高到低的 P_1、P_2、P_3 这 3 个作业。它们使用设备的先后顺序和占用设备时间分别如下。

作业 P_1：IO2(30ms)→CPU(10ms)→IO1(30ms)→CPU(10ms)

作业 P_2：IO1(20ms)→CPU(20ms)→IO2(40ms)

作业 P_3：CPU(30ms)→IO1(20ms)

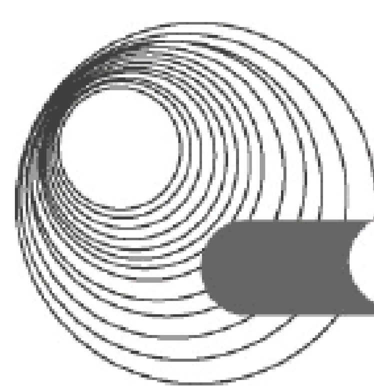
在对于其他辅助操作时间可以忽略不计的假设下，作业 P_1、P_2、P_3 从投入到完成所用的时间分别是 (1) ms、 (2) ms 和 (3) ms。3 个作业从投入运行到全部完成，CPU 的利用率约为 (4)%，IO1 的利用率约为 (5)%。

假定在系统中仅有这 3 个作业投入运行，各设备的利用率是指该设备的使用时间同作业进程全部完成所占用最长时间的比率。

(1)～(5) A．60　　B．67　　C．70　　D．78　　E．80
F．89　　G．90　　H．100　　I．110　　J．120

4.6.4　同步练习参考答案

(1) E　　(2) G　　(3) G　　(4) D　　(5) D



4.7 网络操作系统和嵌入式操作系统实例

4.7.1 考点辅导

4.7.1.1 网络操作系统

1. 网络操作系统的基本概念

计算机网络系统除了硬件，还需要有系统软件，两者结合构成计算机网络的基础平台。操作系统是最重要的软件。网络操作系统是网络用户和计算机网络之间的一个接口，它除了应具备通常操作系统应具备的基本功能外，还应有联网功能，支持网络体系结构和各种网络通信协议，提供网络互联功能，支持有效、可靠、安全的数据传送。

典型的网络操作系统的特征如下。

(1) 硬件独立性。网络操作系统可以运行在不同的网络硬件上，可以通过网桥或路由器与别的网络连接。

(2) 多用户支持。应能同时支持多个用户对网络的访问，应对信息资源提供完全的安全和保护功能。

(3) 支持网络实用程序及其管理功能，如系统备份、安全管理、容错和性能控制。

(4) 多种客户端支持。如 Windows NT 网络操作系统包括 OS/2、Windows 98、UNIX 等多种客户端，极大地方便了网络用户。

(5) 提供目录服务。以单一逻辑的方式让用户访问位于世界范围内的所有网络服务和资源的技术。

(6) 支持多种增值服务，如文件服务、打印服务、通信服务和数据库服务等。

2. 网络操作系统的类型

一般来说，网络操作系统可以分为以下 3 类。

(1) 集中模式。集中式网络操作系统是由分时操作系统加上网络功能演变而来的，系统的基本单元是由一台主机和若干台与主机相连的终端构成，将多台主机连接起来形成了网络，信息的处理和控制是集中的。UNIX 就是这类系统的典型例子。

(2) 客户机/服务器模式。客户机/服务器模式是流行的网络工作模式。该种模式的网络可分为两个部分，即服务器和客户机。服务器是网络的控制中心，其任务是向客户机提供一种或多种服务。服务器可有多种类型，如提供文件或打印服务的文件服务器等。客户机是用于本地处理和进行服务器访问的站点，在客户机中包含本地处理软件和访问服务器上服务程序的软件接口。

(3) 对等模式。采用这种模式的操作系统网络中，各个站点是对等的。它既可以作为客户去访问其他站点，又可以作为服务器向其他站点提供服务；在网络中既无服务处理中心，也无控制中心，或者说网络的服务和控制功能分布在各个站点上。

3. 常见的网络操作系统

目前，流行的网络操作系统主要有以下几种。

(1) Microsoft 公司的 Windows NT Server 操作系统。

(2) Novell 公司的 NetWare 操作系统。

(3) IBM 公司的 LANServer 操作系统。

(4) UNIX 操作系统。

(5) Linux 操作系统。

4.7.1.2 嵌入式操作系统

1. 嵌入式操作系统的概念

嵌入式操作系统是指在嵌入式系统中的操作系统。嵌入式操作系统运行在嵌入式智能芯片环境中，对整个智能芯片以及它所操作、控制的各种部件装置等资源进行统一协调、调度、指挥和控制。

2. 嵌入式操作系统的特点

嵌入式操作系统具有占用空间小、执行效率高、方便进行个性化定制和软件要求固化存储等特点。

(1) 微型化。由于硬件平台的局限性，如主存少、字长短、运行速度有限、能源少(用微小型电池)、外部设备和控制对象千变万化，因此，不论从性能还是从成本角度考虑，都不允许它占用很多资源，系统代码量少，应在保证应用功能的前提下，以微型化作为特点来设计嵌入式操作系统的结构与功能。

(2) 可定制。嵌入式操作系统的运行平台多种多样，应用更是千变万化，表现出专业化的特点。从减少成本和缩短研发周期方面考虑，要求它能运行在不同的微处理机平台上，能针对硬件变化进行结构与功能上的配置，以满足不同应用的需要。

(3) 实时性。嵌入式操作系统广泛应用于过程控制、数据采集、传输通信、多媒体信息及一些需要迅速响应的场合，实时响应要求严格，因此实时性是其主要特点之一。

(4) 可靠性。系统构件、模块和体系结构必须达到应有的可靠性，对关键要害应用还要提供容错和防故障措施，以进一步提高可靠性。

(5) 易移植性。为了提高系统的易移植性，通常采用硬件抽象层(Hardware Abstraction Level，HAL)和板级支撑包(Board Support Package，BSP)的底层设计技术。HAL 提供了与设备无关的特性，屏蔽了硬件平台的细节和差异，向操作系统上层提供了统一接口，保证了系统的可移植性。

3. 嵌入式系统开发环境

嵌入式系统开发环境通常配有源码级可配置的系统模块设计、丰富的同步原语、可选择的调度算法、可选择的主存分配策略、定时器与计数器、多方式中断处理支持、多种异常处理选择、多种通信方式支持、标准 C 语言库、数学运算库和开放式应用程序接口等。较著名的嵌入式操作系统有 Windows CE、VxWorks、pSOS、Palm OS 等。

4.7.2 同步练习

网络操作系统主要可以分为 3 类，即集中模式、客户机/服务器模式和_____。

A．对等模式　　B．交互模式　　C．多用户模式　　D．多客户端模式

4.7.3 同步练习参考答案

A

4.8 UNIX 操作系统实例

4.8.1 考点辅导

4.8.1.1 UNIX 操作系统

UNIX 操作系统是一种多用户、多任务的分时操作系统。它由最内层的硬件提供基本服务，内核提供全部应用程序所需的各种服务。

4.8.1.2 UNIX 文件系统

UNIX 文件系统采用树形带交叉勾连的目录结构，根目录即为“/”，非叶节点是目录文件，叶节点可以是目录文件，也可以是文件或特殊文件。目录是一个包含目录项的文件，在逻辑上可以认为每一个目录项都包含一个文件名，同时还包含说明该文件属性的信息。某些 UNIX 文件系统限制文件名的最大长度为 14 个字符，BSD 版本则将这种限制扩展到了 255 个字符。

UNIX 文件系统结构由四部分组成，即引导块、超级块、索引节点区和数据存储区。

- 引导块：占据文件系统的开头，通常占用一个物理块，包含引导代码段。
- 超级块：描述文件系统的状态。
- 索引节点区：第一个索引节点就是文件系统的根索引节点，当执行 mount 命令后，该文件系统的目录结构就可以从这个根索引节点开始进行存取。
- 数据存储区：专门存放数据的区域。

4.8.1.3 UNIX 进程与存储管理

1．进程管理

UNIX 中的进程由进程控制块(PCB)、正文段和数据段组成。PCB 由常驻主存的基本进程控制块 proc 和非常驻主存的进程扩充控制块 user 两部分组成。正文段可供多个进程共享。系统设置一正文表 text，每个正文段都占据一个表项，用来指明正文段在主存和磁盘中的位置。数据段是进程执行时用到的数据段，若进程执行时的程序是非共享的，则也构成数据

段的一部分。

传统的 UNIX 进程控制子系统有进程同步、进程通信、存储管理和进程调度几大功能。进程调度采用动态优先数调度算法，进程的优先数随着进程的执行情况而变化。UNIX 系统中优先数的确定方法有两种，即设置方法和计算方法。

2. 存储管理

UNIX 早期的版本采用“对换技术”扩充主存容量，进程可以被换出到对换区，也可以从对换区换进到主存。高版本的 UNIX 主存管理采用分页式虚拟存储机制，对换技术作为一种辅助手段，并采用二次机会页面替换算法。

4.8.1.4 UNIX 设备管理

1. 设备管理

UNIX 系统中的文件等同于系统中可用的任何资源。UNIX 的设计者们遵循一条这样的规则：UNIX 系统中可以使用的任何计算机资源都用一种统一的方法表示。它们选择用“文件”这个概念作为一切资源的抽象表示方法。

UNIX 系统包括两类设备，即块设备和字符设备。UNIX 设备管理的主要特点如下。

(1) 块设备与字符设备具有相似的层次结构。这是指对它们的控制方法和所采用的数据结构、层次结构几乎相同。

(2) 将设备作为一个特殊文件，并赋予一个文件名。这样，对设备的使用类似于对文件的存取，具有统一的接口。

(3) 采用完善的缓冲区管理技术。引入“预先读”“异步写”和“延迟写”方式，进一步提高了系统效率。

2. 输入输出转向

在 UNIX 中，任何一个存放一条或多条命令的文件称为 Shell 程序或 Shell 过程。Shell 向用户提供了输入输出的转向命令，可以在不改变应用程序本身的情况下自由地改变其数据的输入源和输出目的地。其中，“>”“>>”表示输出转向，“<”表示输入转向。

3. 管道

在 UNIX 系统中，“|”符号表示管道。一个管道总是连接两条命令。若将左边的标准输出命令和右边的标准输入命令相连，则左边命令的输出结果就直接成为右边命令的输入。这个功能使得用户可以在不改动程序本身的前提下使多个程序通过标准输入输出设备进行数据传递。

4.8.1.5 shell 程序

shell 不但负责管理命令行界面，而且 shell 自己也是一个编程的环境。实际上，可以将命令按照命令行的格式写入一个文件，再将其权限设置为可执行，就可以像普通命令一样执行它了。这个文件通常称为脚本。熟悉 DOS 的用户自然想到 shell 脚本相当于 DOS 的批处理文件，而且 shell 脚本中也同样支持如 if、for 和 case 等程序控制流程，甚至还支持变量和函数定义。shell 实际上是一种编程语言。利用 shell 语言可以编写出功能很强的 shell 程序，并可将程序段组合起来。

1. 正则表达式

在 UNIX 中，正则表达式不仅用在 vi 中，还用在 shell 中。正则表达式是用来确定字符串模式的一个规则集，是对文本字符串的一种描述，该描述能简洁而又完整地刻画文本字符串的关键特性。因此，正则表达式通常被用作字符串的匹配操作。

2. shell 变量

shell 中的变量可分为 3 种类型，即用户定义变量、系统定义变量和 shell 定义变量。

用户定义变量是必须以字符或下划线开始，可以包含字母、下划线和数字的字符序列。系统定义变量和 shell 定义变量要相对复杂一些。

3. shell 程序

shell 向用户提供了许多用于简化输入的符号，这些符号包括各种通配符、字符串定义符、转义符和变量定义符等。这些符号可以被看作 shell 的保留字，通常称为“元字符”。元字符的种类和作用非常多，它们无论在 shell 的命令行输入还是在 shell 程序设计中都起着非常重要的作用。

shell 命令行本身也是一个交互式的脚本执行环境，也就是说，在命令行上同样可以使用脚本中的控制语句，也可以定义变量(实际上就是环境变量)，甚至可以定义函数。这都与脚本文件中的命令一样。但是有一点必须注意，shell 程序有许多种，不同的 shell 有不同的编程命令和语法。虽然它们基本上大同小异，但还是有许多差别。

4.8.2 典型例题分析

例 1 UNIX 系统采用直接、一级、二级和三级间接索引技术访问文件，其索引节点有 13 个地址项(i_addr[0]～i_addr[12])。如果每个盘块的大小为 1 KB，每个盘块号占 4B，则进程 A 访问文件 F 中第 11264 字节处的数据时，(28)。(2009 年下半年试题 28)

(28) A. 可直接寻址　　B. 需要一次间接寻址
C. 需要二次间接寻址　　D. 需要三次间接寻址

解析：由于 UNIX 系统可以提供 4 级索引——10 个直接索引块、1 个一次间接索引、1 个二次间接索引、1 个 3 次间接索引。当一个进程要访问的偏移量为 11264B 时，需要访问磁盘的次数计算如下。

偏移量为 11264B 在文件中的相对块号和块内字节为 11246/1024=11。

每个盘块号占 4B，一个索引块可以存放 1024 个索引项。

显然，11 块在一次间接索引块中，且占有的索引项为 11-10=1。故只需要一次间接寻址，以 1 为索引找到相对块对应的物理块。

答案：B

例 2 在 UNIX 操作系统中，把输入输出设备看作 (23) 。(2006 年上半年试题 23)

(23) A. 普通文件　B. 目录文件　C. 索引文件　D. 特殊文件

解析：UNIX 系统将外部设备作为文件看待，称为特殊文件(或特别文件、设备文件)，由文件系统统一管理。这样可以使输入/输出硬件的接口与普通文件接口一致。

答案：D

4.8.3　同步练习

1. 在 UNIX 操作系统中，把输入输出设备看作__(1)__。
   A．普通文件　　B．目录文件　　C．索引文件　　D．特殊文件
2. 在 UNIX 操作系统中，当用户执行以下命令：

```
link("/user/include/myfile.sh", "/usr/userwang/youfile.sh")
```

则文件名“/usr/userwang/youfile.sh”存放在__(22)__。(2003 年试题 22)
   A．user 目录文件中　　B．include 目录文件中
   C．userwang 目录文件中　　D．youfile.sh 的文件内容中

4.8.4　同步练习参考答案

1．D　2．C

4.9　本 章 小 结

本章知识点在 2013 年的新大纲有一定改动，主要在存储管理这个知识点中新增加了分段存储管理和段页式存储管理两个考点，去掉了操作系统实例中的 Windows 2000/XP 操作系统的相关知识。另外，还有一些描述方面的调整。

本章主要要求考生掌握操作系统的基本概念，包括操作系统的内核、处理机管理、存储管理、设备管理、文件管理、作业管理和用户界面，以及网络操作系统和嵌入式操作系统的基础知识及其实例。其中作业管理以及网络操作系统和嵌入式操作系统基础知识，从 2009—2013 年这 8 次考试中没有出现过考题，但是这些内容还是大纲要求掌握的内容，建议考生掌握相应的知识点。

这一章在历年考试中所占的比分比较稳定，一般都是考 6 道或者 7 道小题。

对操作系统的学习关键是要建立整体观念，充分掌握系统活动的机理，以常用的典型系统为主线，抓住重点。

4.10　达标训练题及参考答案

4.10.1　达标训练题

1．在操作系统中，进程是一个具有一定独立功能的程序在某个数据集合上的一次__(1)__。进程是一个__(2)__的概念，而程序是一个__(3)__的概念。

在一单处理机中，若有 5 个用户进程，在非管态的某一时刻，处于就绪状态的用户进程最多有 (4) 个，最少有 (5) 个。

(1) A. 并发活动　B. 运行活动　C. 单独操作　D. 关联操作

(2)～(3) A. 组合态　B. 关联态　C. 运行态　D. 等待态　E. 静态　F. 动态

(4)～(5) A. 1　B. 2　C. 3　D. 4　E. 5　F. 0

2．从下面有关进程的概念和性质叙述中，选出 5 条正确的叙述，(1)、(2)、(3)、(4)、(5)，并把编号按从小到大的次序写在答卷的(1)～(5)栏内。

A．唤醒：挂起→就绪　B．封锁：就绪→挂起

C．调度：就绪→运行　D．超时：运行→挂起

E．超时：运行→就绪　F．用户进程可激发调度进程

G．用户进程可激发唤醒进程　H．用户进程可激发超时进程

I．进程接近 CPU 可接纳的限度时，可降低页面出错的频率

J．进程具有引用局部性时，可降低页面出错的频率

3．虚拟存储管理系统的基础是程序的 (1) 理论。这个理论的基本含义是指程序执行时往往会 (2) 访问主存储器。程序的 (1) 表现在 (3) 和 (4) 上。(3) 是指最近被访问的存储单元可能马上又要被访问。(4) 是指马上被访问的单元，而其附近的单元也可能马上被访问。

根据这个理论，Denning 提出了工作集理论。工作集是进程运行时被频繁访问的页面集合。在进程运行时，如果它的工作页面都在 (5) 内，能够使该进程有效地运行；否则会出现频繁的页面调入/调出现象。

(1) A．局部性　B．全局性　C．动态性　D．虚拟性

(2) A．频繁地　B．均匀地　C．不均匀地　D．全面地

(3)～(4) A．数据局部性　B．空间局部性　C．时间局部性
D．数据全局性　E．空间全局性　F．时间全局性

(5) A．外部存储器　B．主存　C．辅助存储器　D．虚拟存储器

4．由于主存大小有限，为使得一个或多个作业能在系统中运行，常需要外存来换取主存。其中以作业为单位对内外存进行交换的技术称为 (1) 技术，在作业内部对内外存进行交换的技术称为 (2) 技术。用外存换主存是以牺牲程序运行时间为代价的。为提高 CPU 的有效利用率，避免内外存的频繁交换，(2) 技术常用某种 (3) 来选择换出主存的页面，它的基础是程序的 (4) 。据此，操作系统可根据 (5) 来改善系统的性能，(5) 是一个进程在定长的执行时间区间内涉及的页面集合。

(1)～(2) A．SPOOLing　B．Swapping
C．虚拟存储　D．虚拟机
E．进程管理　F．设备管理

(3) A．页面分配策略　B．页面标志策略
C．页面淘汰策略　D．段设置策略

(4) A．完整性　B．局部性　C．递归性　D．正确性

(5) A．工作集　B．页面集　C．段号集　D．元素集

4.10.2　参考答案

1.	(1) B	(2) F	(3) E	(4) D	(5) F
2.	(1) A	(2) C	(3) E	(4) G	(5) J
3.	(1) A	(2) C	(3) C	(4) B	(5) B
4.	(1) B	(2) C	(3) C	(4) B	(5) A

第 5 章　软件工程基础知识

大纲要求：

- 软件工程、软件过程改进和软件开发项目管理知识，包括软件工程知识，软件开发生命周期各阶段的目标和任务，软件开发项目管理基础知识(时间管理、成本管理、质量管理、人力资源管理、风险管理等)及其常用管理工具，主要的软件开发方法(生命周期法、原型法、面向对象法、CASE)，软件开发工具与环境知识，软件过程改进知识，软件质量管理知识，软件开发过程评估、软件能力成熟评估基础知识。
- 诊断与容错技术。
- 风险分析、风险类型、抗风险措施和内部控制。
- 系统分析基础知识，包括系统分析的目的和任务、结构化分析方法(数据流图(DFD)、数据字典(DD)、实体关系图(ERD)、描述加工处理的结构化语言)、统一建模语言(UML)、系统规格说明书。
- 系统设计知识，包括系统设计的目的和任务、结构化设计方法和工具(系统流程图、HIPO 图、控制流程图)、系统总体结构设计(总体布局、设计原则、模块结构设计、数据存储设计、系统配置方案)、系统详细设计(代码设计、数据库设计、用户界面设计、处理过程设计)、系统设计说明书。
- 系统实施知识，包括系统实施的主要任务，结构化程序设计、面向对象程序设计、可视化程序设计，程序设计风格，程序设计语言的选择，系统测试的目的、类型，系统测试方法(黑盒测试、白盒测试、灰盒测试)，测试设计和管理(错误曲线、错误排除、收敛、注入故障、测试用例设计、系统测试报告)，系统转换基础知识。
- 系统运行和维护知识，包括系统运行管理基础知识、系统维护基础知识、系统评价基础知识。

5.1　软件工程概述

5.1.1　考点辅导

5.1.1.1　计算机软件

计算机软件是指计算机系统中的程序及其文档。程序是计算任务的处理对象和处理规则的描述。任何以计算机为处理工具的任务都是计算任务。处理对象是数据(如数字、文字、图形、图像、声音等，它们只是表示，而无含义)或信息(数据及有关的含义)。处理规则一般指处理的动作和步骤。文档是为了便于了解程序所需的阐述性资料。

按照软件的应用领域，可以将计算机软件分为十大类：系统软件、应用软件、工程 / 科学软件、嵌入式软件、产品线软件、Web 应用、人工智能软件、开放计算、网络资源和开源软件。

5.1.1.2　软件工程基本原理

美国著名的软件工程专家 B.W.Boehm 于 1983 年提出了软件工程的 7 条基本原理。Boehm 认为这 7 条原理是确保软件产品质量和开发效率的原理的最小集合。

1. 用分阶段的生命周期计划严格管理

有统计表明，50070 以上的失败项目是由于计划不周造成的。在软件开发与维护的漫长生命周期中，需要完成许多各种各样的工作。这条基本原理意味着应该把软件生命周期划分成若干个阶段，并相应地制订出切实可行的计划，然后严格按照计划对软件的开发与维护工作进行管理。Boehm 认为，在软件的整个生存周期中应该制定并严格执行六类计划：项目概要计划、里程碑计划、项目控制计划、产品控制计划、验证计划和运行维护计划。

2. 坚持进行阶段评审

据统计结果显示，大部分错误是在编码之前造成的。根据 Boehm 等人的统计，设计错误占软件错误的 63070，编码错误仅占 37%，而且错误发现与改正得越晚，所需付出的代价越高。因此，在每个阶段都应进行严格的评审，以便尽早发现在软件开发过程中所犯的错误。

3. 实现严格的产品控制在软件开发过程中不应随意改变需求，因为改变一项需求需要付出较高的代价

但是，在软件开发过程中改变需求又是难免的，由于外部环境的变化，相应地改变用户需求是一种客观需要，这就要采用科学的产品控制技术来顺应这种要求。在改变需求时，为了保持软件各个配置成分的一致性，必须实行严格的产品控制，其中主要是实行基准配置管理。基准配置又称为基线配置，它是经过阶段评审后的软件配置成分(各个阶段产生的文档或程序代码)。基准配置管理也称为变动控制，一切有关修改软件的建议，特别是涉及基准配置的修改建议，都必须按照严格的规程进行评审，在获得批准以后才能实施修改。

4. 采用现代程序设计技术

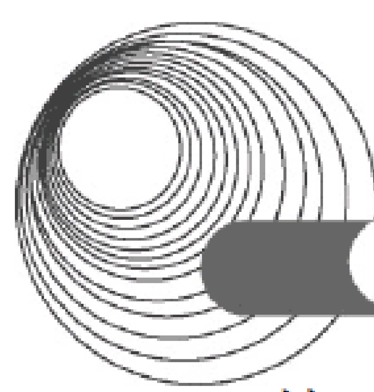
从 20 世纪 60 年代和 70 年代的结构化软件开发技术到面向对象技术，从第一代、第二代语言到第四代语言，人们已经充分认识到：方法大于力气。采用先进的技术既可以提高软件开发的效率，又可以降低软件维护的成本。

5. 结果应能清楚地审查

软件是一种看不见、摸不着的逻辑产品。软件开发小组的工作进展情况可见性差，难以评价和管理。为了更好地进行管理，应根据软件开发的总目标及完成期限尽量明确地规定开发小组的责任和产品标准，从而使所得到的结果能够清楚地审查。

6. 开发小组的人员应少而精

开发人员的素质和数量是影响软件质量和开发效率的重要因素，应该少而精。这一条



基于两点原因：高素质开发人员的效率比低素质开发人员的效率要高几倍到几十倍，开发工作中犯的错误也要少得多；当开发小组为 N 人时，可能的通信信道为 N(N-I)/2。可见，随着人数 N 的增大，通信开销将急剧增大。

7. 承认不断改进软件工程实践的必要性

遵循上述 6 条基本原理，就能够按照当代软件工程基本原理实现软件的工程化生产。但是它们只是对现有经验的总结和归纳，并不能保证软件开发与维护的过程能赶上时代前进的步伐，能跟上技术的不断进步。因此，Boehm 提出应把“承认不断改进软件工程实践的必要性”作为软件工程的第 7 条原理。根据这条原理，用户不仅要积极采纳新的软件开发技术，还要注意不断总结经验，收集进度和消耗等数据，进行出错类型和问题报告统计。这些数据既可以用来评估新的软件技术的效果，也可以用来指明必须着重注意的问题和应该优先进行研究的工具和技术。

5.1.1.3 软件生存周期

同任何事物一样，软件也有一个孕育、诞生、成长、成熟、衰亡的生存过程，被称为计算机软件的生存周期。通常，软件生存周期包括可行性分析与项目开发计划、需求分析、设计(概要设计和详细设计)、编码、测试、维护等阶段。

(1) 可行性分析与项目开发计划。这个阶段主要确定软件的开发目标及其可行性。必须要回答的问题是：要解决的问题是什么？该问题有可行的解决办法吗？若有解决的办法，需要多少费用？需要多少资源？需要多少时间？要回答这些问题，就要进行问题定义、可行性分析，制订项目开发计划。

可行性分析与项目计划阶段的参加人员有用户、项目负责人和系统分析师。该阶段所产生的文档有可行性分析报告和项目开发计划。

(2) 需求分析。需求分析阶段的任务不是具体地解决问题，而是准确地确定软件系统必须做什么，确定软件系统的功能、性能、数据和界面等要求，从而确定系统的逻辑模型。该阶段的参加人员有用户、项目负责人和系统分析师。产生的文档有软件需求说明书。

(3) 概要设计。概要设计就是设计软件的结构，明确软件由哪些模块组成，这些模块的层次结构是怎样的，这些模块的调用关系是怎样的，每个模块的功能是什么。同时，还要设计该项目的应用系统的总体数据结构和数据库结构，即应用系统要存储什么数据、这些数据是什么样的结构，它们之间有什么关系。概要设计阶段参加的人员有系统分析师和软件设计师。该阶段的主要文档有概要设计说明书。

(4) 详细设计。详细设计阶段的主要任务就是对每个模块完成的功能进行具体描述，要把功能描述转变为精确的、结构化的过程描述。即该模块的控制结构是怎样的，先做什么，后做什么，有什么样的条件判定，有哪些要重复处理等，并用相应的表示工具把这些控制结构表示出来。详细设计阶段参加的人员有软件设计师和程序员。该阶段的主要文档有详细设计文档。

(5) 编码。编码阶段就是把每个模块的控制结构转换成计算机可接受的程序代码，即写成某种特定程序设计语言表示的源程序清单。

(6) 测试。测试是保证软件质量的重要手段，其主要方式是在设计测试用例的基础上检查软件的各个组成部分。测试阶段的参加人员通常由另一部门(或单位)的软件设计师或系

统分析师承担。该阶段产生的文档有软件测试计划、测试用例和软件测试报告。

(7) 维护。软件维护是软件生存周期中时间最长的阶段。已交付的软件投入正式使用后，便进入软件维护阶段，它可以持续几年甚至几十年。软件运行过程中可能由于各方面的原因，需要对它进行修改，也就是维护的过程。

5.1.1.4　软件过程

在开发产品或构建系统时，遵循一系列可预测的步骤(即路线图)是非常重要的，它有助于及时交付高质量的产品。软件开发中所遵循的路线图称为“软件过程”。过程是活动的集合，活动是任务的集合。软件过程有 3 层含义：一个是个体含义，即指软件产品或系统在生存周期中的某一类活动的集合，如软件开发过程、软件管理过程等；二是整体含义，即指软件产品或系统在所有上述含义下的软件过程的总体；三是工程含义，即指解决软件过程的工程，应用软件的原则、方法来构造软件过程模型，并结合软件产品的具体要求进行实例化，以及在用户环境下的运作，以此进一步提高软件的生产率，降低成本。

1. 能力成熟度模型(CMM)

CMM 将软件组织的过程能力分成五个成熟度级别：初始级、可重复级、已定义级、已管理级和优化级。由低到高，软件开发生产精度越来越高，每单位工程的生产周期越来越短。

(1) 初始级。软件过程是无序的，有时甚至是混乱的，对过程几乎没有定义，成功取决于个人努力。

(2) 可重复级。建立了基本的项目管理过程来跟踪费用、进度和功能特性；制定了必要的过程纪律，能重复早先类似应用项目取得的成功。

(3) 定义级。已将软件管理和工程两方面的过程文档化、标准化，并综合成该组织的标准软件过程。所有项目均使用经批准、剪裁的标准软件过程来开发和维护软件。

(4) 管理级。收集对软件过程和产品质量的详细度量，对软件过程和产品都有定量的理解和控制。

(5) 优化级。过程的量化反馈和先进的新思想、新技术促使过程不断改进。

2. 能力成熟度模型集成(CMMI)

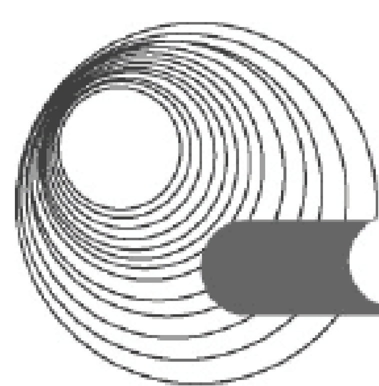
CMM 的成功导致了适用不同学科领域的模型的衍生，如系统工程的能力成熟度模型，适用于集成化产品开发的能力成熟度模型等。而一个工程项目又往往涉及多个交叉的学科，因此有必要将各种过程改进的工作集成起来。1998 年，由美国产业界、政府和卡内基•梅隆大学软件工程研究所共同主持 CMMI 项目。CMMI 是若干过程模型的综合和改进，是支持多个工程学科和领域的、系统的、一致的过程改进框架，能适应现代工程的特点和需要，能提高过程的质量和工作效率。

CMMI 提供了两种表示方法：阶段式模型和连续式模型。

1) 阶段式模型

阶段式模型的结构类似于 CMM，它关注组织的成熟度。CMMI-SE/SW/IPPD 1.1 版中有 5 个成熟度等级。

初始的：过程不可预测且缺乏控制。



已管理的：过程为项目服务。

已定义的：过程为组织服务。

定量管理的：过程已度量和控制。

优化的：集中于过程改进。

2) 连续式模型

连续式模型关注每个过程域的能力，一个组织对不同的过程域可以达到不同的过程域能力等级(Capability Level，CL)。CMMI 中包括 6 个过程域能力等级，等级号为 0-5。能力等级包括共性目标及相关的共性实践，这些实践在过程域内被添加到特定目标和实践中。当组织满足过程域的特定目标和共性目标时，就说该组织达到了那个过程域的能力等级。

能力等级可以独立地应用于任何单独的过程域，任何一个能力等级都必须满足比它等级低的能力等级的所有准则。对各能力等级的含义简述如下。

CLo(未完成的)：过程域未执行或未得到 CLi 中定义的所有目标。

CLi(已执行的)：其共性目标是过程将可标识的输入工作产品转换成可标识的输出工作产品，以实现支持过程域的特定目标。

CL2(已管理的)：其共性目标集中于已管理的过程的制度化。根据组织级政策规定过程的运作将使用哪个过程，项目遵循已文档化的计划和过程描述，所有正在工作的人都有权使用足够的资源，所有工作任务和工作产品都被监控、控制和评审。

CL3(已定义级的)：其共性目标集中于已定义的过程的制度化。过程是按照组织的剪裁指南从组织的标准过程集中剪裁得到的，还必须收集过程资产和过程的度量，并用于将来对过程的改进。

CL4(定量管理的)：其共性目标集中于可定量管理的过程的制度化。使用测量和质量保证来控制和改进过程域，建立和使用关于质量和过程执行的定量目标作为管理准则。

CLs(优化的)：使用量化(统计学)手段改变和优化过程域，以满足客户要求的改变和持续改进计划中的过程域的功效。

5.1.2 典型例题分析

软件工程的基本要素包括方法、工具和<u>(15)</u>。(2015 年上半年试题 15)

(15) A．软件系统　　B．硬件系统　　C．过程　　D．人员

解析：软件工程包括三个要素：方法、工具和过程。软件工程方法为软件开发提供了“如何做”的技术。它包括了多方面的任务，如项目计划与估算、软件系统需求分析、数据结构、系统总体结构的设计、算法过程的设计、编码、测试以及维护等。软件工具为软件工程方法提供了自动的或半自动的软件支撑环境。过程定义了方法使用的顺序、要求交付的文档资料、为保证质量和协调变化所需要的管理及软件开发各个阶段完成的里程碑。

答案：C

5.1.3 同步练习

基于构件的软件开发，强调使用可复用的软件“构件”来设计和构建软件系统，对所

需的构件进行合格性检验、(1)，并将它们集成到新系统中。

(15) A．规模度量　　B．数据验证　　C．适应性修改　　D．正确性测试

5.1.4　同步练习参考答案

答案：C

5.2　软件过程模型

5.2.1　考点辅导

软件过程模型习惯上也称为软件开发模型，它是软件开发全部过程、活动和任务的结构框架。典型的软件过程模型有瀑布模型、增量模型、演化模型(原型模型、螺旋模型)、喷泉模型、基于构件的开发模型和形式化方法模型等。

5.2.1.1　瀑布模型

该模型给出了软件生存周期各阶段的固定顺序，上一阶段完成后才能进入到下一阶段，整个过程就像流水下泻，故称为瀑布模型。瀑布模型为软件的开发和维护提供一种有效的管理模式，对保证软件产品的质量有重要的作用。但是这种模型缺乏灵活性，无法通过开发活动来澄清本来不够明确的需求，这将可能导致直到软件开发完成时才发现所开发的软件并非都是用户所需要的，此时必须付出高额的代价才能纠正这一偏差。

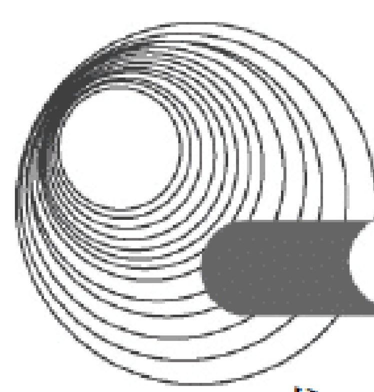
5.2.1.2　增量模型

增量模型融合了瀑布模型的基本成分和原型实现的迭代特征，它假设可以将需求分段为一系列增量产品，每一增量可以分别开发。该模型采用随着日程时间的进展而交错的线性序列，每一个线性序列产生软件的一个可发布的“增量”。当使用增量模型时，第 1 个增量往往是核心的产品。客户对每个增量的使用和评估都作为下一个增量发布的新特征和功能，这个过程在每一个增量发布后不断重复，直到产生了最终的完善产品。增量模型强调每一个增量均发布一个可操作的产品。

5.2.1.3　演化模型

在获取一组基本的需求后，通过快速分析构造出该软件的一个初始可运行版本，这个初始的软件通常称为原型，然后根据用户在使用原型的过程中提出的意见和建议对原型进行改进，获得原型的新版本。重复这一过程，最终可得到令用户满意的软件产品。该模型是用于对软件需求缺乏准确认识的情况。

(1)　原型模型。原型是预期系统的一个可执行版本，反映了系统性质的一个选定的子集。一个原型不必满足目标软件的所有约束，其目的是能快速、低成本地构建原型。当然，能够采用原型方法是因为开发工具的快速发展，使得能够迅速地开发出一个让用户看得见、摸得着的系统框架。这样，对于计算机不是很熟的用户就可以根据这个框架提出自己的需



求。开发原型系统首先确定用户需求，开发初始原型，然后征求用户对初始原型的改进意见，并根据意见修改原型。

(2) 螺旋模型。将瀑布模型和演化模型相结合就成了螺旋模型。这种模型综合了瀑布模型和演化模型的优点，并增加了风险分析。螺旋模型包括四个方面的活动：制订计划、风险分析、实施工程、客户评估。

5.2.1.4 喷泉模型

喷泉模型主要用于描述面向对象的开发过程。该模型具有迭代和无间隙特性。迭代意味着模型中的开发活动常常需要重复多次，在迭代中不断完善软件系统。无间隙是指在开发活动之间不存在明显的边界，允许开发活动交叉、迭代地进行。

5.2.1.5 基于构件的模型

基于构件的开发是指利用预先包装的构件来构造应用系统。构件可以是组织内部开发的构件，也可以是商品化成品(Commercial Off-The-Shelf，COTS)软件构件。基于构件的开发模型具有许多螺旋模型的特点，它本质上是演化模型，需要以迭代方式构建软件。其不同之处在于，基于构件的开发模型采用预先打包的软件构件开发应用系统。

5.2.1.6 形式化方法模型

形式化方法是建立在严格数学基础上的一种软件开发方法，其主要活动是生成计算机软件形式化的数学规格说明。

形式化方法用严格的数学语言和语义描述功能规约和设计规约，通过数学的分析和推导，易于发现需求的歧义性、不完整性和不一致性，易于对分析模型、设计模型和程序进行验证。通过数学的演算，使得从形式化功能规约到形式化设计规约，以及从形式化设计规约到程序代码的转换成为可能。这种方法的一个变形是净室软件工程。

5.2.1.7 统一过程(UP)模型

统一过程(UP)模型是一种“用例和风险驱动，以架构为中心，迭代并且增量”的开发过程，由 UML 方法和工具支持。迭代的意思是将整个软件开发项目划分为许多个小的“袖珍项目”，每个“袖珍项目”都包含正常软件项目的所有元素，如计划、分析和设计、构造、集成和测试以及内部和外部发布。

统一过程包括 5 个阶段，即初始阶段、精化阶段、构建阶段、移交阶段、产生阶段。前 4 个技术阶段由主要里程碑所终止。

(1) 初始阶段——生命周期目标。

(2) 精化阶段——生命周期架构。

(3) 构建阶段——初始运作功能。

(4) 移交阶段——产品发布。

统一过程的典型代表是 RUP(Rational Unified Process)。RUP 是 UP 的商业扩展，完全兼容 UP，但比 UP 更完整、更详细。

5.2.1.8 敏捷方法

敏捷开发的总体目标是通过“尽可能早地、持续地对有价值的软件的交付”使客户满意。

(1)　极限编程(XP)。XP 是一种轻量级(敏捷)、高效、低风险、柔性、可预测的、科学的软件开发方式。它由价值观、原则、实践和行为 4 个部分组成，彼此相互依赖、关联，并通过行为贯穿于整个生存周期。

4 个价值观，包括沟通、简单性、反馈和勇气。

5 个原则，包括快速反馈、简单性假设、逐步修改、提倡更改和优质工作。

12 个最佳实践，包括计划游戏(快速制订计划、随着细节的不断变化而完善)、小型发布(系统的设计要能够尽可能早地交付)、隐喻(找到合适的比喻传达信息)、简单设计(只处理当前的需求，使设计保持简单)、测试先行(先写测试代码，然后再编写程序)、重构(重新审视需求和设计，重新明确地描述它们以符合新的和现有的需求)、结队编程、集体代码所有制、持续集成(可以按日甚至按小时为客户提供可运行的版本)、每周工作 40h、现场客户和编码标准。

(2)　水晶法。

(3)　并列争球法。

(4)　自适应软件开发。

5.2.2　典型例题分析

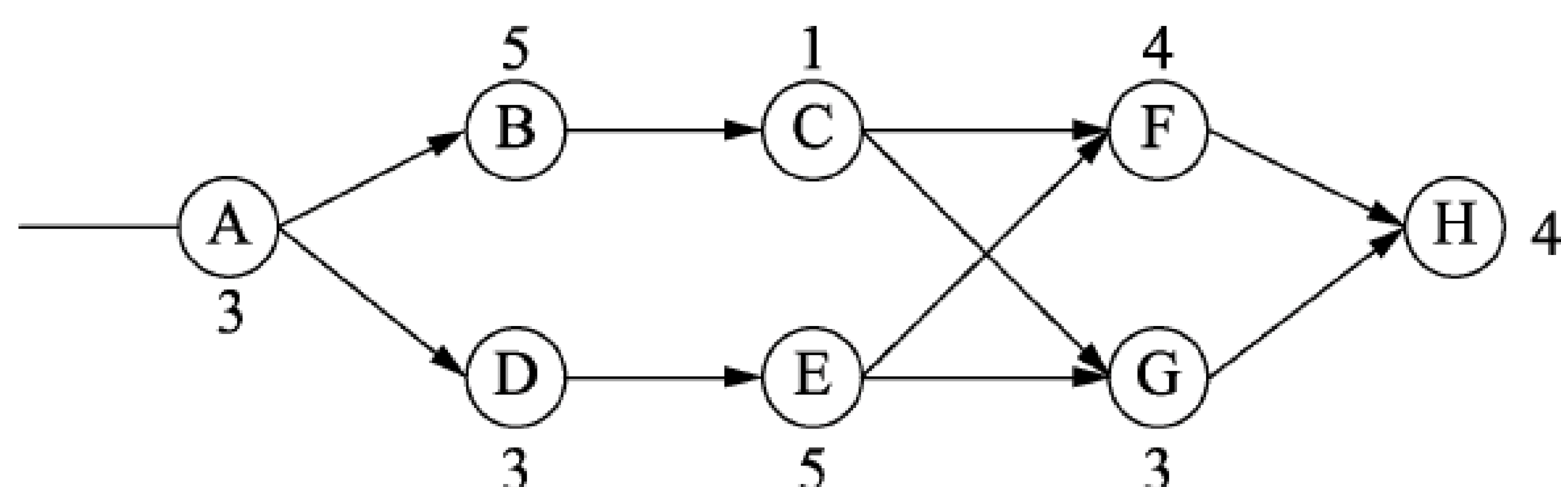
例 1　某项目包含的活动如下表所示，完成整个项目的最短时间为 (17) 周。不能通过缩短活动 (18) 的工期，来缩短整个项目的完成时间。(2015 年上半年试题 17、18)

活动编号	工期/周	直接前驱
A	3	—
B	5	A
C	1	B
D	3	A
E	5	D
F	4	C，E
G	3	C，E
H	4	F，G

(17)　A. 16　　B. 17　　C. 18　　D. 19

(18)　A. A　　B. B　　C. D　　D. F

解析：根据活动表绘制活动图，可知关键路径为 A→D→E→F→H，长度为 19，所以最短工期 19 周。由于 B 不是关键路径上的活动，所以压缩 B 无法缩短整个项目的完成时间。



答案：(17) D　　(18) B

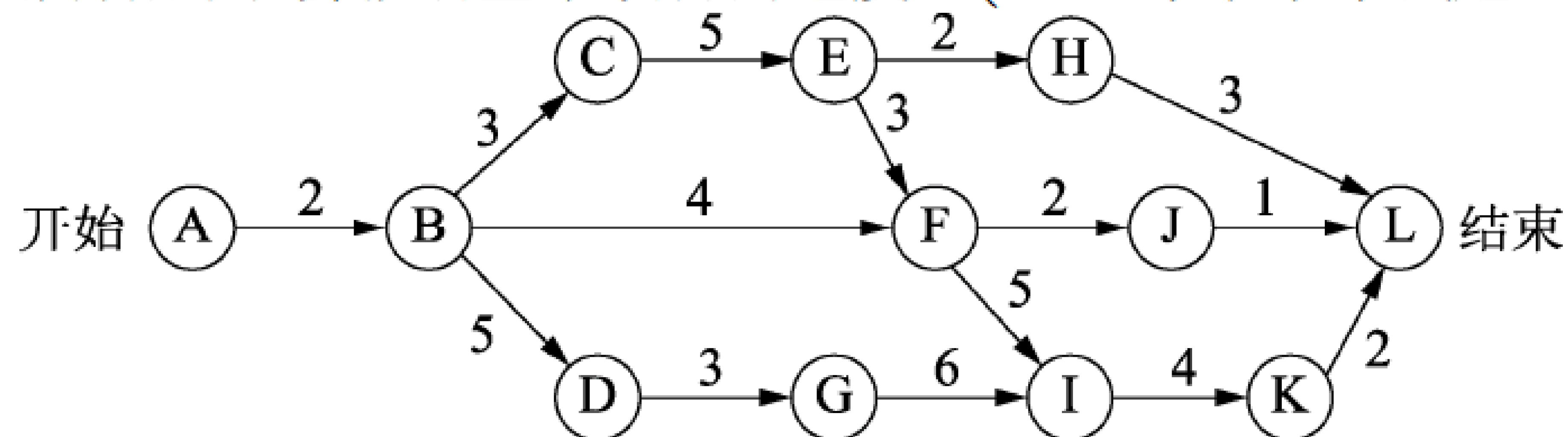
例 2 在<u>(16)</u>设计阶段选择适当的解决方案，将系统分解为若干个子系统，建立整个系统的体系结构。(2015 年上半年试题 16)

(16) A．概要 B．详细 C．结构化 D．面向对象

解析：概要设计就是要设计软件的结构，明确软件由哪些模块组成，这些模块的层次结构如何，这些模块是如何调用的，以及每个模块的功能。同时，还要设计该项目的应用系统的总体数据结构和数据库结构。

答案：A

例 3 某软件项目的活动图如下图所示，其中顶点表示项目里程碑，连接顶点的边表示活动，边上的数字表示该活动所需的天数，则完成该项目的最少时间为<u>(17)</u>天。活动 BD 最多可以晚<u>(18)</u>天开始而不会影响整个项目的进度。(2015 年下半年试题 17、18)



(17) A．9 B．15 C．22 D．24

(18) A．2 B．3 C．5 D．9

解析：本题关键路径为：A→B→C→E→F→I→K→L，是活动图中花费时间最长的活动的序列，长度为 24。BD 在路径 A→B→D→G→I→K→L，长度为 22，比关键路径短 2，因此，要想不影响整个项目的进度，活动 BD 最多可以晚 2 天开始。

答案：(17) D (18) A

例 4 若用户需求不清晰且经常发生变化，但系统规模不太大且不太复杂，则最适宜采用<u>(15)</u>开发方法；对于数据处理领域的问题，若系统规模不太大且不太复杂，需求变化也不大，则最适宜采用<u>(16)</u>开发方法。(2015 年下半年试题 15、16)

(15) A．结构化 B．Jackson C．原型化 D．面向对象

(16) A．结构化 B．Jackson C．原型化 D．面向对象

解析：结构化方法的指导思想是自顶向下、逐层分解，特别适用于数据处理领域的问题，不适合解决大规模的、特别复杂的项目，难以适应需求的变化。Jackson 方法是以数据结构为驱动的，适合于小规模的项目，适用于时序特点较强的系统。原型化方法适合于用户需求不清、需求经常变化的情况，系统规模不大也不太复杂时采用该方法较好。面向对象方法尽可能按照人类认识世界的方法和思维方式来分析和解决问题，可用于用户需求经常变化、规模复杂的项目，在人工智能、并发工程和综合集成工程等方面得到了广泛的应用。

答案：(15)C (16)A

例 5 在屏蔽软件错误的容错系统中，冗余附加技术的构成不包括<u>(32)</u>。(2012 年下半年试题 32)

(32) A．关键程序和数据的冗余存储及调用

B．冗余备份程序的存储及调用

C．实现错误检测和错误恢复的程序

D. 实现容错软件所需的固化程序

解析：冗余附加技术是指为实现冗余技术所需的资源和技术，包括程序、指令、数据、存放和调动它们的空间和通道等。在屏蔽硬件错误的容错技术中，冗余附加技术包括关键程序和数据的冗余存储及调用，检测、表决、切换、重构、纠错和复算的实现。在屏蔽软件错误的容错系统中，冗余附加技术的构成包括冗余备份程序的存储及调用、实现错误检测和错误恢复的程序、实现容错软件所需的固化程序。

答案：A

例 6　以下关于结构化开发方法的叙述中，不正确的是 (15) 。(2014 年下半年试题 15)

(15) A. 总的指导思想是自顶向下，逐层分解

B. 基本原则是功能的分解与抽象

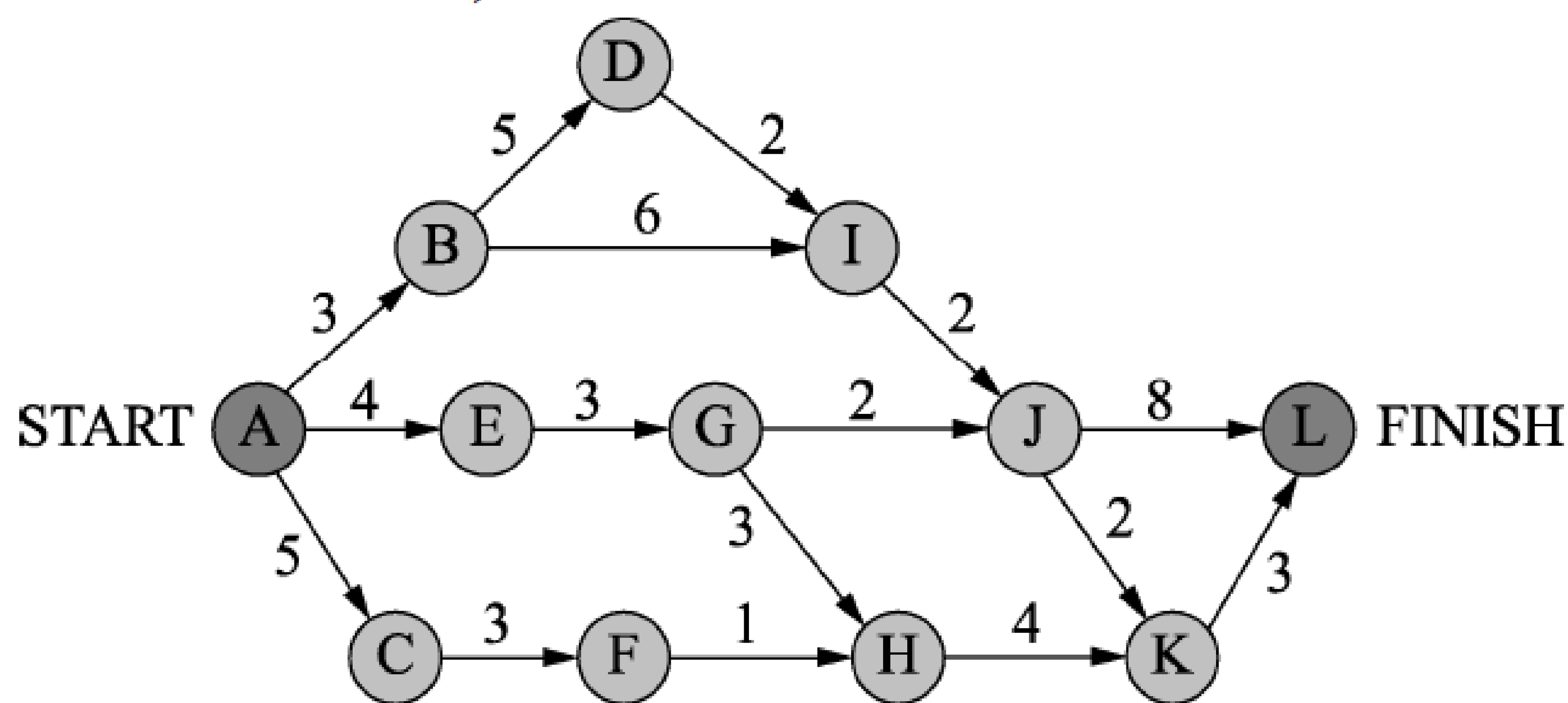
C. 与面向对象开发方法相比，更适合大规模、特别复杂的项目

D. 特别适合于数据处理领域的项目

解析：结构化系统开发方法(Structured System Development Methodology)是目前应用得最普遍的一种开发方法。其基本思想是用系统的思想和系统工程的方法，按照用户至上的原则结构化、模块化，自顶向下对系统进行分析与设计。它不限于数据处理领域项目。

答案：C

例 7　下图是一个软件项目的活动图，其中顶点表示项目里程碑，连接顶点的边表示活动，边的权重表示活动的持续时间，则里程碑 (19) 在关键路径上。活动 GH 的松弛时间是 (20) 。(2014 年下半年试题 19、20)



(19) A. B　　B. E　　C. C　　D. K

(20) A. 0　　B. 1　　C. 2　　D. 3

解析：关键路径为 A-B-D-I-J-L

GH 活动最早开始时间是 7；最晚开始时间是 10，所以松弛时间是 3。

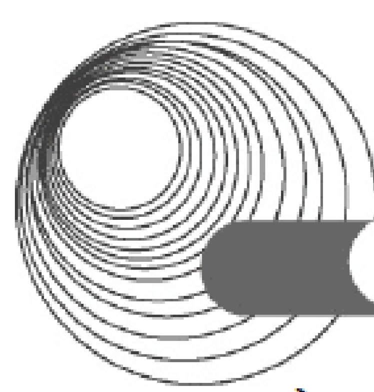
答案：(19) A　　(20) D

例 8　以下关于增量模型的叙述中，正确的是 (29) 。(2014 年下半年试题 29)

(29) A. 需求被清晰定义　　B. 可以快速构造核心产品

C. 每个增量必须要进行风险评估　　D. 不适宜商业产品的开发

解析：增量模型融合了瀑布模型的基本成分(重复应用)和原型实现的迭代特征，该模型采用随着日程时间的进展而交错的线性序列，每一个线性序列产生软件的一个可发布的“增量”。当使用增量模型时，第 1 个增量往往是核心的产品，即第 1 个增量实现了基本的需求，但很多补充的特征还没有发布。客户对每一个增量的使用和评估都作为下一个增量发布的新特征和功能，这个过程在每一个增量发布后不断重复，直到产生了最终的完善



产品。

答案：B

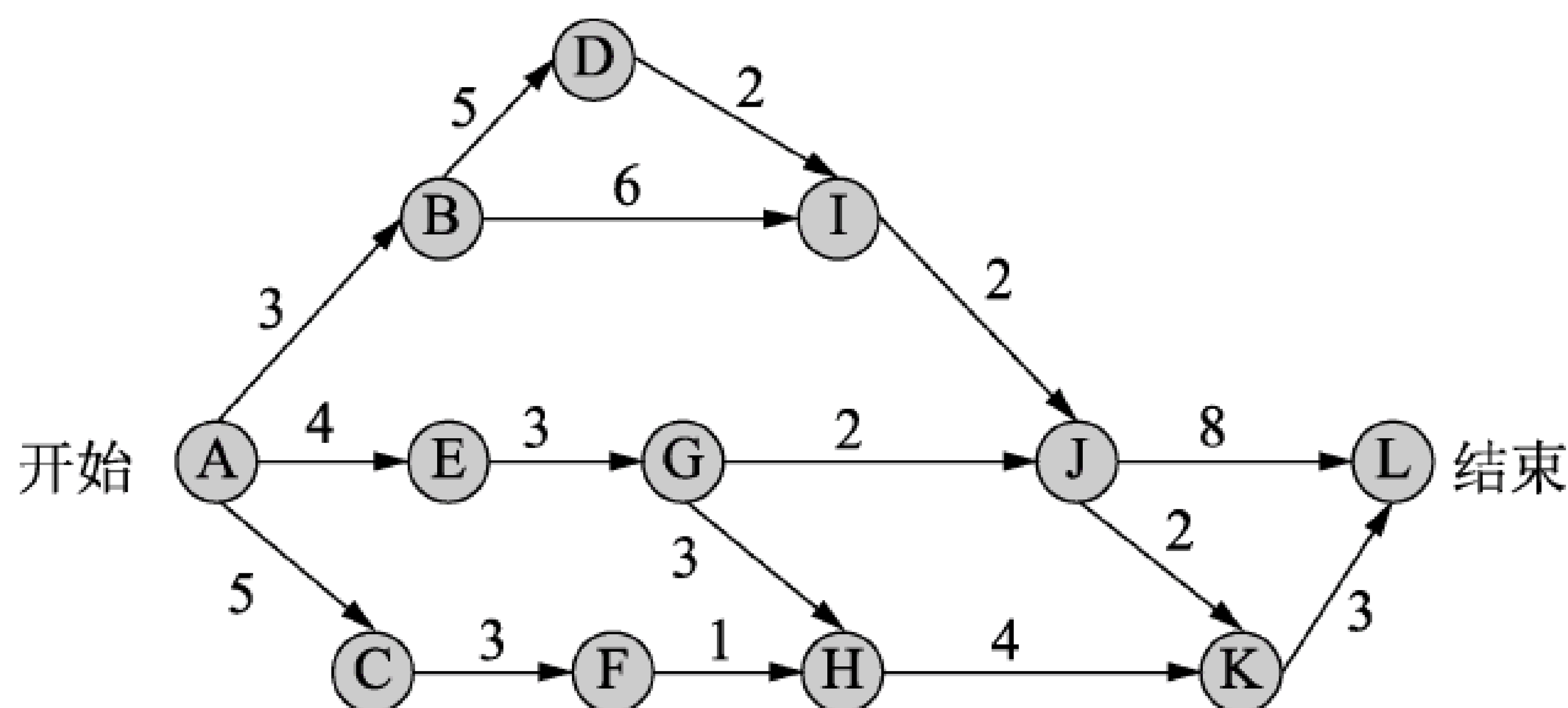
例 9 某开发小组欲开发一个超大规模软件，即使用通信卫星，在订阅者中提供、监视和控制移动电话通信，则最不适宜采用 (29) 过程模型。(2015 年下半年试题 29)

(29) A. 瀑布　　B. 原型　　C. 螺旋　　D. 喷泉

解析：瀑布模型假设，一个待开发的系统需求是完整的、简明的、一致的，而且可以先于设计和实现完成之前产生，适用于需求明确的软件项目。本题中所要开发的是超大规模软件，需求不明确，不适宜采用瀑布模型。

答案：A

例 10 下图是一个软件项目的活动图，其中顶点表示项目里程碑，连接顶点的边表示包含的活动，边上的值表示完成活动所需要的时间，则关键路径长度为 (17) 。(2011 年下半年试题 17)



(17) A. 20　　B. 19　　C. 17　　D. 16

解析：关键路径为 A→B→D→I→J→L，因为这条路径最长，决定了整个项目完成的最短时间为 20。

答案：A

例 11 (30) 开发过程模型以用户需求为动力，以对象为驱动，适合于面向对象的开发方法。(2015 年下半年试题 30)

(30) A. 瀑布　　B. 原型　　C. 螺旋　　D. 喷泉

解析：瀑布模型规定了软件开发和维护的前后顺序，是面向过程的方法；原型模型和螺旋模型都是演化模型，是迭代的过程模型。

答案：D

例 12 某公司计划开发一种产品，技术含量很高，与客户相关的风险也很多，则最适于采用 (29) 开发过程模型。(2015 年上半年试题 29)

(29) A. 瀑布　　B. 原型　　C. 增量　　D. 螺旋

解析：这些模型中仅有螺旋模型考虑风险因素。螺旋模型将瀑布模型和演化模型结合起来，加入了两种模型均忽略的风险因素，弥补了这两种模型的不足。由于螺旋模型强化了风险因素，使得开发人员和用户对每个演化层出现的风险有所了解，从而作出应有的反应。因此，该模型特别适合于庞大、复杂且有高风险的系统。

答案：D

例 13 将每个用户的数据和其他用户的数据隔离开，是考虑了软件的(31)质量特性。

(2011 年下半年试题 31)

(31) A. 功能性　　B. 可靠性　　C. 可维护性　　D. 易使用性

解析：“将每个用户的数据和其他用户的数据隔离开”是从安全性方面提出的功能要求，这主要是考虑了软件的功能性质量特性。

答案：A

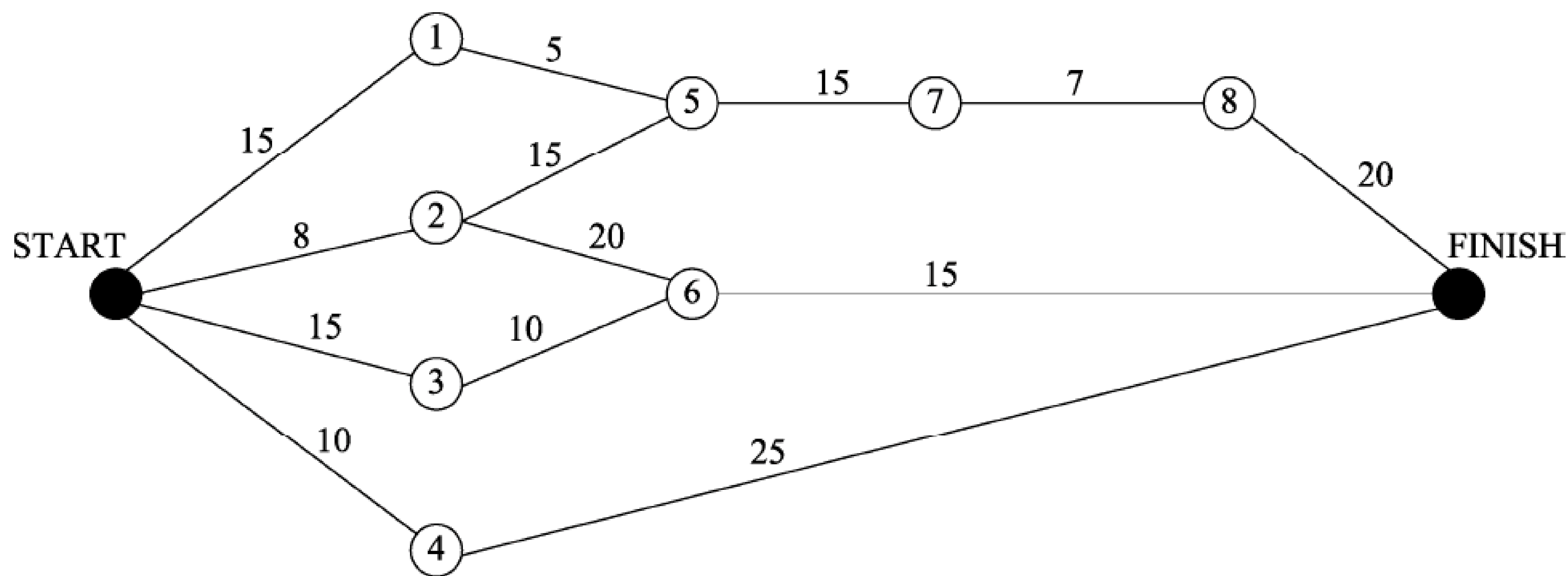
例 14　包含 8 个成员的开发小组的沟通路径最多有 (15) 条。(2011 年上半年试题 15)

(15) A. 28　　B. 32　　C. 56　　D. 64

解析：需要协作沟通的人员的数量影响着开发成本，因为成本的主要组成部分是相互的沟通和交流，以及更正沟通不当所引起的不良结果。人与人之间必须通过沟通来解决各自承担任务之间的接口问题，如果项目有 n 个工作人员，则有 $n(n-1)/2$ 个相互沟通的路径。很明显，包含 8 个成员的开发小组的沟通路径最多有 28 条。这其实是一道简单的图论问题，相当于求包含 8 个顶点的无向图中最多有多少条边。

答案：A

例 15　下图是一个软件项目的活动图，其中顶点表示项目里程碑，边表示包含的活动，边上的权重表示活动的持续时间，则里程碑 (19) 在关键路径上。(2011 年上半年试题 19)



(19) A. 1　　B. 2　　C. 3　　D. 4

解析：本题的关键路径是：开始→2→5→7→8→结束，因为这一条路径最长，决定了整个项目完成的最短时间为 65。很明显，里程碑 2 在关键路径上。

答案：B

例 16　以下关于 CMM 的叙述中，不正确的是 (31) 。(2014 年下半年试题 31)

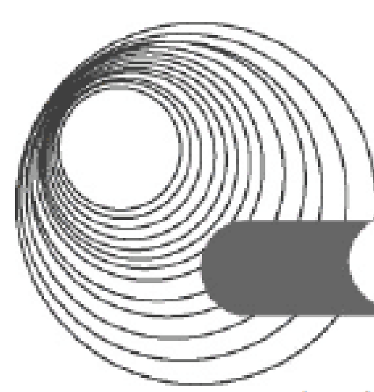
(31) A. CMM 是指软件过程能力成熟度模型

B. CMM 根据软件过程的不同成熟度划分了 5 个等级，其中，1 级被认为成熟度最高，5 级被认为成熟度最低

C. CMMI 的任务是将已有的几个 CMM 模型结合在一起，使之构造成为“集成模型”

D. 采用更成熟的 CMM 模型，一般来说可以提高最终产品的质量

解析：CMM 模型指“能力成熟度模型”，其英文全称为 Capability Maturity Model for Software，英文缩写为 SW-CMM，简称 CMM。它是对于软件组织在定义、实施、度量、控制和改善其软件过程的实践中各个发展阶段的描述。CMM 的核心是把软件开发视为一个过程，并根据这一原则对软件开发和维护进行过程监控和研究，以使其更加科学化、标准化、



使企业能够更好地实现商业目标。

CMM模型分为5个等级：1. 初始级；2. 可重复级；3. 已定义级；4. 定量管理级；5. 优化级。

答案：B

例17 某公司要开发一个软件产品，产品的某些需求是明确的，而某些需求则需要进一步细化。由于市场竞争的压力，产品需要尽快上市，则开发该软件产品最不适合采用 (30) 模型。(2014年上半年试题30)

(30) A. 瀑布　　B. 原型　　C. 增量　　D. 螺旋

解析：瀑布模型将软件生命周期划分为需求分析、软件设计、程序编写、软件测试和运行维护等基本活动，并且规定了它们自上而下、相互衔接的固定次序，如同瀑布流水，逐级下落。瀑布模型要求待开发的系统需求是完整的、简明的、一致的，其突出的缺点是不适应用户需求的变化，开发模型是线性的，用户只有等到整个过程的末期才能见到开发成果。对于需求不确定的软件产品不适合采用瀑布模型。

答案：A

例18 敏捷开发方法XP是一种轻量级、高效、低风险、柔性、可预测、科学的软件开发方法，其特性包含在12个最佳实践中。系统的设计要能够尽可能早交付，属于 (18) 最佳实践。(2010年下半年试题18)

(18) A. 隐喻　　B. 重构　　C. 小型发布　　D. 持续集成

解析：12个最佳实践如下。

- 计划游戏(快速制订计划、随着细节的不断变化而完善)。
- 小型发布(系统的设计要能够尽可能早地交付)。
- 隐喻(找到合适的比喻传达信息)。
- 简单设计(只处理当前的需求，使设计保持简单)。
- 测试先行(先写测试代码，然后再编写程序)。
- 重构(重新审视需求和设计，重新明确地描述它们以符合新的和现有的需求)。
- 结队编程。
- 集体代码所有制。
- 持续集成(可以按日甚至按小时为客户提供可运行的版本)。
- 每周工作40h。
- 现场客户。
- 编码标准。

答案：C

例19 风险的优先级通常是根据 (19) 设定。(2015年上半年试题19)

(19) A. 风险影响(Risk Impact)　　B. 风险概率(Risk Probability)
C. 风险暴露(Risk Exposure)　　D. 风险控制(Risk Control)

解析：风险优先级是按风险影响大小排出一个风险优先级列表，这个列表将作为风险控制的基础。风险分析要评估每一个风险出现的可能性及其影响，判定风险的级别。风险暴露是一种很有用的风险分析方法，它是以风险发生的概率乘以损失的程度作为风险暴露量。举例来说，如果你认为“完成需求分析比原计划延长4周的概率是30%”，那么风险

暴露量就是 4 周×30%=1.2 周。

答案：C

例 20　以下关于过程改进的叙述中，错误的是__(30)__。(2010 年下半年试题 30)

(30) A．过程能力成熟度模型基于这样的理念：改进过程将改进产品，尤其是软件产品

B．软件过程改进框架包括评估、计划、改进和监控四个部分

C．软件过程改进不是一次性的，需要反复进行

D．在评估后要把发现的问题转化为软件过程改进计划

解析：过程能力成熟度模型的基本思想是，由于问题是由人们管理软件过程的方法不当引起的，所以新软件技术的运用并不会自动提高软件的生产率和质量。其策略是力图改进对软件过程的管理，而在技术方面的改进是其必然的结果。

答案：B

5.2.3　同步练习

1.　以下关于统一过程 UP 的叙述中，不正确的是__(1)__。

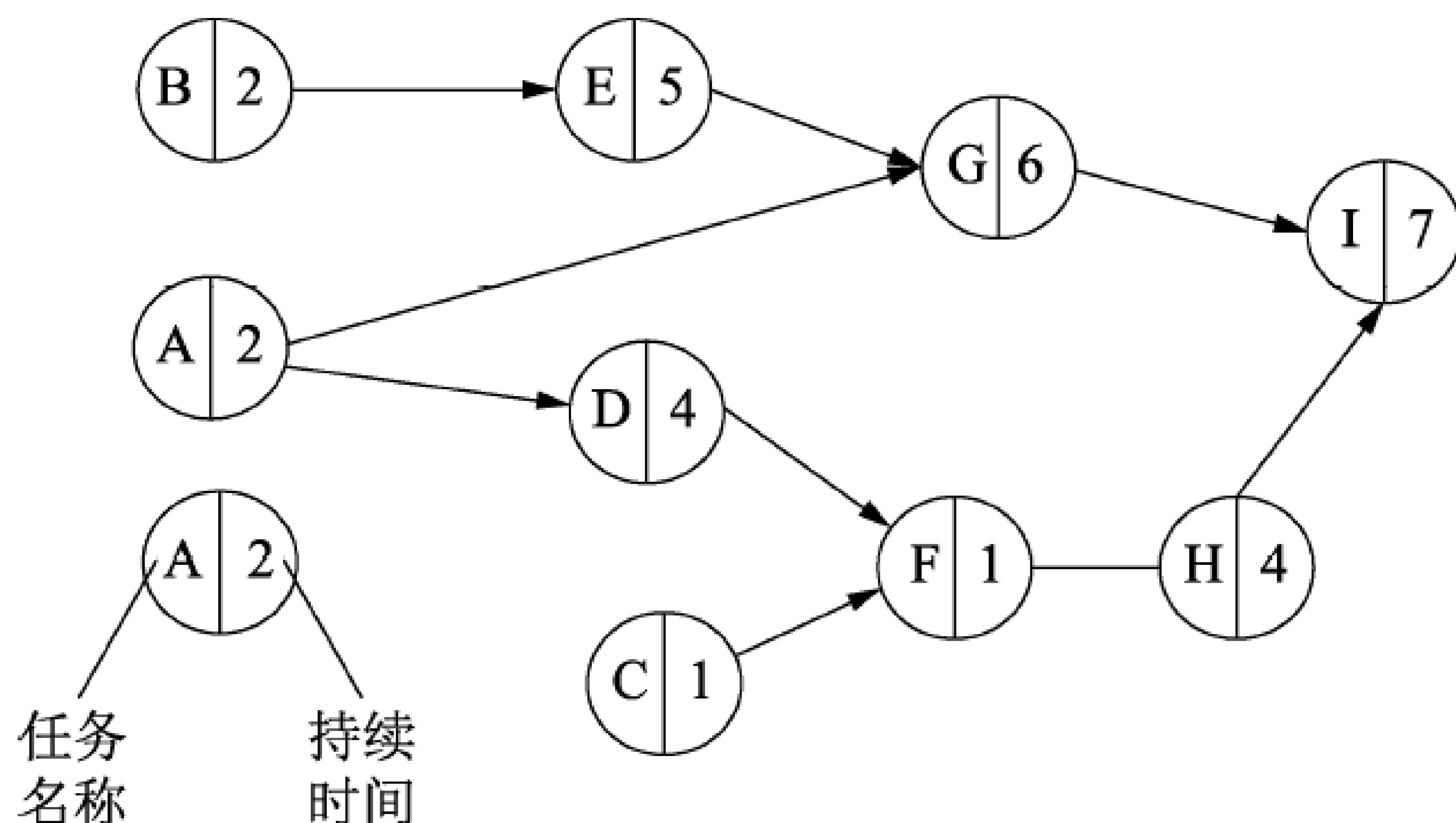
(1) A．UP 是以用例和风险为驱动，以架构为中心，迭代并且增量的开发过程

B．UP 定义了四个阶段，即起始、精化、构建和确认阶段

C．每次迭代都包含计划、分析、设计、构造、集成、测试以及内部和外部发布

D．每个迭代有五个核心工作流

2.　某项目主要由 A～I 任务构成，其计划图展示了各任务之间的前后关系以及每个任务的工期(单位：天)，该项目的关键路径是__(2)__。在不延误项目总工期的情况下，任务 A 最多可以推迟开始的时间是__(3)__天。



(2) A．A→G→I　　B．A→D→F→H→I

C．B→E→G→I　　D．C→F→H→I

(3) A．0　　B．2　　C．5　　D．7

3.　项目复杂性、规模和结构的不确定性属于__(4)__风险。

(4) A．项目　　B．技术　　C．经济　　D．商业

4.　软件能力成熟度模型(CMM)将软件能力成熟度自低到高依次划分为 5 级。目前，达到 CMM 第三级(已定义级)是许多组织努力的目标，该级的核心是__(5)__。(2009 年上半年试题 29)

(5) A. 建立基本的项目管理和实践来跟踪项目费用、进度和功能特性
B. 使用标准开发过程(或方法论)构建(或集成)系统
C. 管理层寻求更主动地应对系统的开发问题
D. 连续地监督和改进标准化的系统开发过程

5. RUP 在每个阶段都有主要目标，并在结束时产生一些制品。在 (6) 结束时产生"在适当的平台上集成的软件产品"。

(6) A. 初级阶段 B. 精化阶段 C. 构建阶段 D. 移交阶段

6. 在 ISO/IEC 软件质量模型中，易使用性的子特性不包括 (7) 。(2015 年下半年试题 31)

(7) A. 易理解性 B. 易学性 C. 易操作性 D. 易分析性

7. 在开发信息系统时，用于系统开发人员与项目管理人员沟通的主要文档是 (8) 。

(8) A. 系统开发合同 B. 系统设计说明书
C. 系统开发计划 D. 系统测试报告

5.2.4 同步练习参考答案

答案：

(1)B (2) C (3) B (4) A (5)B (6)C (7) D (8)C

5.3 需 求 分 析

5.3.1 考点辅导

5.3.1.1 软件需求

在进行需求获取之前，首先要明确需要获取什么，也就是需求包含哪些内容。软件需求是指用户对目标软件系统在功能、行为、性能、设计约束等方面的期望。通常，这些需求包括功能需求、性能需求、用户或人的因素、环境需求、界面需求、文档需求、数据需求、资源使用需求、安全保密需求、可靠性需求、软件成本消耗与开发进度需求等，并预先估计以后系统可能达到的目标。此外，还需要注意其他非功能性的需求。具体内容如下。

(1) 功能需求。
(2) 性能需求。
(3) 用户或人的因素。
(4) 环境需求。
(5) 界面需求。
(6) 文档需求。
(7) 数据需求。
(8) 资源使用需求。

(9) 安全保密要求。
(10) 可靠性要求。
(11) 软件成本消耗与开发进度需求。
(12) 其他非功能性要求。

1. 需求分析的任务

需求分析主要是确定待开发软件的功能、性能、数据、界面等要求。具体来说有下面几点。

(1) 确定软件系统的综合要求，包括系统界面、功能、性能、安全性、保密性、可靠性、运行等方面的要求。

(2) 分析软件系统的数据要求，包括基本数据元素、数据元素之间的逻辑关系、数据量、峰值等。

(3) 导出系统的逻辑模型，在结构化方法中可用数据流图来描述；在面向对象分析方法中可以用类模型来描述。

(4) 修正项目开发计划。

(5) 如有必要，可开发一个原型系统以验证用户的需求。

2. 软件需求的分类

下面介绍软件需求的分类。

(1) 功能需求。所开发的软件必须具备什么样的功能。

(2) 非功能需求。它是指产品必须具备的属性或品质，如可靠性、性能响应时间、容错性和可扩展性等。

(3) 设计约束。其也称为限制条件、补充规约，这通常是对解决方案的一些约束说明。

3. 软件需求分析方法

需求分析方法由对软件的数据域和功能域的系统分析过程及其表示方法组成。它定义了表示系统逻辑视图和物理视图的方式。大多数的需求分析方法是由数据驱动的，数据域具有数据流、数据内容和数据结构 3 种属性。通常一种需求分析方法总要利用其中一种或几种属性。

5.3.1.2　需求分析原则

需求分析过程的具体实现有不同的分析方法，这些方法有自己独特的特点。然而，这些分析方法都遵循一组操作原则。

(1) 必须能够表示和理解问题的信息域。

(2) 必须能够定义软件将完成的任务。

(3) 必须能够表示软件的行为(作为外部事件的结束)。

(4) 必须划分描述数据、功能和行为的模型，从而可以分层次地揭示细节。

(5) 分析过程应该从要素信息移向细节信息。

通过应用这些原则，分析人员将能系统地处理问题。检查信息域可以更完整地理解功能，通过模型可以更简洁地交流功能和行为的特征，应用抽象与分解可减少问题的复杂度。

5.3.1.3　需求工程

需求工程是一个不断反复的需求定义、文档记录、需求演进的过程，并最终在验证的

基础上冻结需求。需求工程可以细分为需求获取、需求分析与协商、系统建模、需求规约、需求验证以及需求管理 6 个阶段。

5.3.2 典型例题分析

为了有效地捕获系统需求，应采用 (29) 。(2011 年上半年试题 29)

(29) A. 瀑布模型　　B. V 模型　　C. 原型模型　　D. 螺旋模型

解析：瀑布模型将软件生命周期的各个活动规定为线性连续的若干阶段模型，它有由前至后、相互衔接的固定次序，适合用于开发需求明确的软件。

V 模型是在瀑布模型的基础上演变而来的，它强调了在整个软件项目开发中需要经历的若干个测试级别，并与每一个开发级别对应。

如果对软件需求的认识是模糊的，可在获取一组基本的需求后，通过快速分析构造出该软件的一个初始可运行版本，即软件的原型，然后根据用户的意见和建议对原型进行改进。

螺旋模型将瀑布模型和原型模型结合起来，加入了风险分析。

题目中的需求还要捕获，也就是说需求还不是很明确，因此适合采用原型模型。

答案：C

5.3.3 同步练习

在软件开发过程中，系统测试阶段的测试目标来自于 (1) 阶段。

(1) A. 需求分析　　B. 概要设计　　C. 详细设计　　D. 软件实现

5.3.4 同步练习参考答案

答案：(1) A

5.4 系 统 设 计

5.4.1 考点辅导

系统设计的主要目的就是为系统制定蓝图，在各种技术和实施方法中权衡利弊，精心设计，合理使用各种资源，最终勾画出新系统的详细设计方案。

5.4.1.1 概要设计

1) 设计软件系统总体结构

设计软件系统总体结构的基本任务是采用某种设计方法，将一个复杂的系统按功能划

分成模块；确定每个模块的功能；确定模块之间的调用关系；确定模块之间的接口，即模块之间传递的信息；评价模块结构的质量。

2)　数据结构及数据库设计

(1) 数据结构的设计。在需求分析阶段，已经通过数据字典对数据的组成、操作约束和数据之间的关系等方面进行了描述，确定了数据的结构特性，在概要设计阶段要加以细化，详细设计阶段则规定具体的实现细节。在概要设计阶段，宜使用抽象的数据类型。

(2) 数据库的设计。数据库的设计是指数据存储文件的设计，主要指以下几个方面。

① 概念设计。在数据分析的基础上，采用自底向上的方法从用户角度进行视图设计，一般用 ER 模型来表述数据模型。

② 逻辑设计。ER 模型是独立于数据库管理系统(DBMS)的，要结合具体的 DBMS 特征来建立数据库的逻辑结构。

③ 物理设计。物理设计就是设计数据模式的一些物理细节，如数据项存储要求、存取方法和索引的建立等。

3)　编写概要设计文档

文档主要有概要设计说明书、数据库设计说明书、用户手册以及修订测试计划。

4)　评审

对设计部分是否完整地实现了需求中规定的功能、性能等要求，设计方法的可行性，关键的处理及内外部接口定义的正确性、有效性以及各部分之间的一致性等都一一进行评审。

5.4.1.2　详细设计

详细设计的基本任务如下。

(1) 对每个模块进行详细的算法设计。用某种图形、表格和语言等工具将每个模块处理过程的详细算法描述出来。

(2) 对模块内的数据结构进行设计。

(3) 对数据库进行物理设计，即确定数据库的物理结构。

(4) 其他设计。根据软件系统的类型，还可能需要进行以下设计。

① 代码设计。为了提高数据的输入、分类、存储和检索等操作，节约内存空间，对数据库中某些数据项的值要进行代码设计。

② 输入输出格式设计。

③ 用户界面设计。

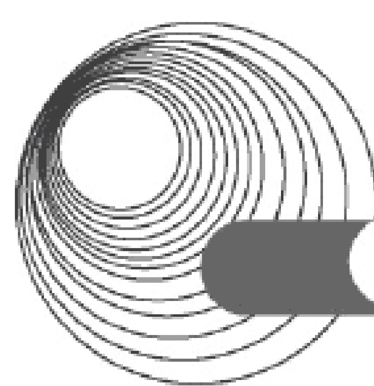
(5) 编写详细设计说明书。

(6) 评审。对处理过程的算法和数据库的物理结构都要评审。

5.4.2　典型例题分析

例 1　__(32)__ 不是属于软件设计质量评审。(2013 年上半年试题 32)

(32) A. 功能与模块之间的对应关系　　B. 软件规格说明是否符合用户的需求
　　 C. 软件是否具有可测试性　　　　D. 软件是否具有良好的可靠性



解析：软件质量评审包括以下内容。

①评价软件的规格说明是否合乎用户的要求，即总体设计思想和设计方针是否正确。②评审可靠性，即是否能避免输入异常(错误或超载等)、硬件失效及软件失效所产生的失效。③评审保密措施实现情况。④评审操作特性实施情况。⑤评审性能实现情况。⑥评审软件是否具有可修改性、可扩充性、可互换性和可移植性。⑦评审软件是否具有可测试性。⑧评审软件是否具有复用性。

因此 A 选项不属于软件质量评审。

答案：A

例 2 模块 A 提供某个班级某门课程的成绩给模块 B，模块 B 计算平均成绩、最高分和最低分，将计算结果返回给模块 A，则模块 B 在软件结构图中属于 (16) 模块。(2014 年上半年试题 16)

(16) A. 传入　　B. 传出　　C. 变换　　D. 协调

解析：模块结构图中的模块类型如下。

(1) 传入模块：从下属模块取得数据，经过某些处理，再将其结果传给上级模块。

(2) 传出模块：从上级模块取得数据，经过某些处理，再将其结果传给下属模块。

(3) 变换模块：也叫加工模块。它从上级模块取得数据，进行特定的处理，转换成其他形式，再传回上级模块。它加工的数据流称为变换数据流。

(4) 协调模块：对所有下属模块进行协调和管理的模块。在系统的输入输出部分或数据加工部分可以找到这样的模块。在一个好的模块结构图中，协调模块应在较高层出现。

答案：C

例 3 某模块实现两个功能：向某个数据结构区域写数据和从该区域读数据。该模块的内聚类型为 (32) 内聚。(2015 年上半年试题 32)

(32) A. 过程　　B. 时间　　C. 逻辑　　D. 通信

解析：过程内聚：一个模块完成多个任务，需要按指定的过程执行。

时间内聚：一个模块完成的功能必须在同一时间间隔内完成(如系统初始化)，但这些功能只是因为时间因素关联在一起。

逻辑内聚：将几个逻辑上相关的功能放在同一模块中。

内聚性：通信内聚>过程内聚>时间内聚>逻辑内聚。

答案：D

例 4 确定软件的规模划分及模块之间的调用关系是 (15) 阶段的任务。(2011 年下半年试题 15)

(15) A. 需求分析　　B. 概要设计　　C. 详细设计　　D. 编码

解析：需求分析阶段的任务是：确定待开发软件的功能、性能、数据和界面要求。

概要设计的任务是：采用某种设计方法，将一个复杂的系统按功能划分成模块；确定每个模块的功能；确定模块之间的调用关系；确定模块之间的接口；评价模块结构的质量。同时还要设计数据结构和数据库。

详细设计的基本任务有：对每个模块进行详细的算法设计；对模块内的数据结构进行设计；对数据库进行物理设计等。

答案：B

例 5　模块 A、B 和 C 包含相同的 5 个语句，这些语句之间没有联系，为了避免重复，把这 5 个模块抽取出来组成模块 D，则模块 D 的内聚类型为 (16) 内聚。(2014 年下半年试题 16)

(16) A．功能　　B．通信　　C．逻辑　　D．巧合

解析：功能内聚：完成一个单一功能，各个部分协同工作，缺一不可。

顺序内聚：处理元素相同，而且必须顺序执行。

通信内聚：所有处理元素集中在一个数据结构的区域上。

过程内聚：处理元素相关，而且必须按特定的次序执行。

瞬时内聚：所包含的任务必须在同一时间间隔内执行(如初始化模块)。

逻辑内聚：完成逻辑上相关的一组任务。

偶然内聚：完成一组没有关系或松散关系的任务。

答案：D

例 6　软件设计时需要遵循抽象、模块化、信息隐蔽和模块独立原则。在划分软件系统模块时，应尽量做到 (30) 。(2010 年上半年试题 30)

(30) A．高内聚高耦合　　B．高内聚低耦合
　　C．低内聚高耦合　　D．低内聚低耦合

解析：耦合性和内聚性是模块独立性的两个定性标准，在划分软件系统模块时，应尽量做到高内聚、低耦合，提高模块的独立性。

答案：B

例 7　面向对象开发方法的基本思想是尽可能按照人类认识客观世界的方法来分析和解决问题， (16) 方法不属于面向对象方法。(2009 年下半年试题 16)

(16) A．Booch　　B．Coad　　C．OMT　　D．Jackson

解析：目前，国际上已经出现多种面向对象的方法，如 Peter Coad 和 Edward Yourdon 的 OOA 和 OOD 方法、Booch 的 OOD 方法、OMT(Object Modeling Technique，面向对象建模技术)方法及 UML(Unified Modeling Language，统一建模语言)。

面向数据结构设计以数据结构作为设计的基础，它根据输入输出数据结构导出程序的结构，适用于规模不大的数据处理系统，Jackson 方法是一种典型的面向数据结构的设计方法。

答案：D

例 8　在进行子系统结构设计时，需要确定划分后的子系统模块结构，并画出模块结构图。该过程不需要考虑 (32) 。(2015 年下半年试题 32)

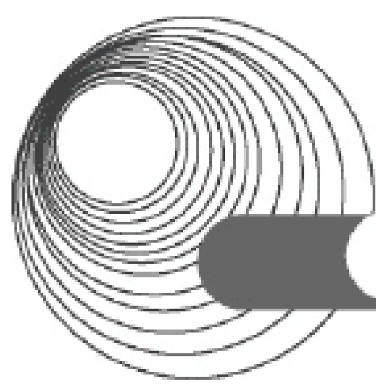
(32) A．每个子系统如何划分成多个模块
　　B．每个子系统采用何种数据结构和核心算法
　　C．如何确定子系统之间、模块之间传送的数据及其调用关系
　　D．如何评价并改进模块结构的质量

解析：子系统结构设计的任务是确定划分后的子系统的模块结构，并画出模块结构图，这个过程中必须考虑以下几个问题。

(1) 每个子系统如何划分成多个模块。

(2) 如何确定子系统之间、模块之间传送的数据及其调用关系。

(3) 如何评价并改进模块结构的质量。



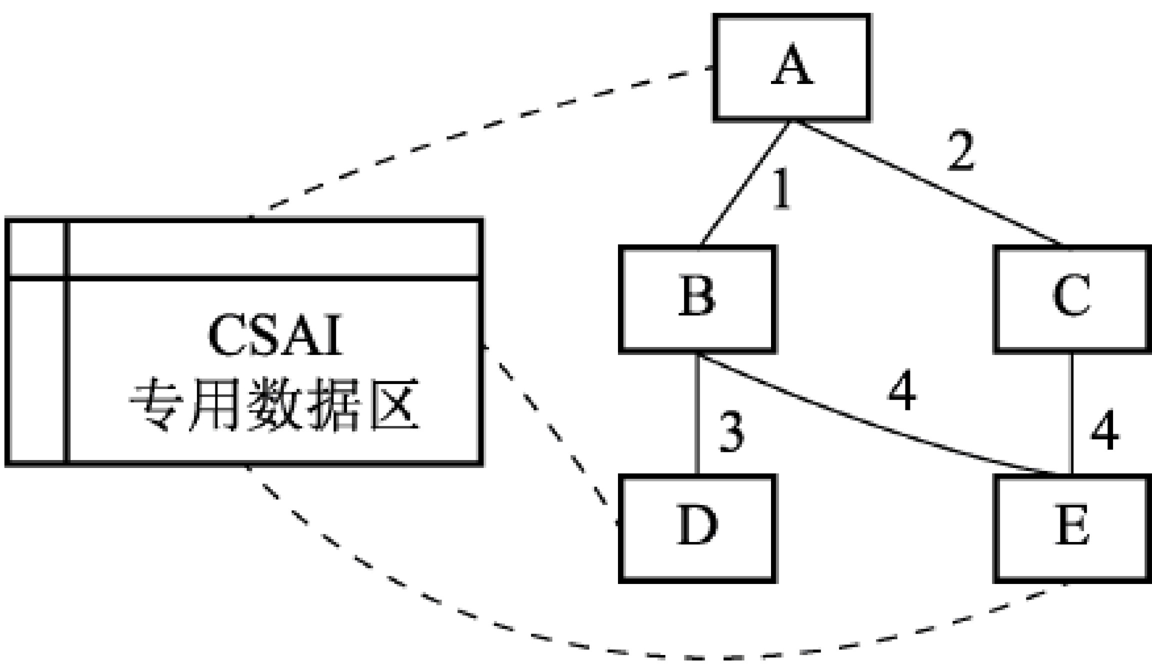
(4) 如何从数据流图导出模块结构图。

答案：B

5.4.3 同步练习

1. 在软件设计和编码过程中，采取________的做法将使软件更加容易理解和维护。
   A. 良好的程序结构，有无文档均可
   B. 使用标准或规定之外的语句
   C. 编写详细正确的文档，采用良好的程序结构
   D. 尽量减少程序中的注释
2. 内聚性和耦合性是度量软件模块独立性的重要准则，软件设计时应力求______。
   A. 高内聚，高耦合　　B. 高内聚，低耦合
   C. 低内聚，高耦合　　D. 低内聚，低耦合
3. 模块的耦合度描述了______。
   A. 模块内各种元素结合的程度　　B. 模块内多个功能之间的接口
   C. 模块之间公共数据的数量　　D. 模块之间相互关联的程度
4. 内聚是一种指标，表示一个模块______。
   A. 代码优化的程度　　B. 代码功能的集中程度
   C. 完成任务的及时程度　　D. 为了与其他模块连接所要完成的工作量

5. 下图中的程序由 A、B、C、D、E 这 5 个模块组成，下表中描述了这些模块之间的接口，每一个接口有一个编号。此外，模块 A、D 和 E 都要引用一个专用数据区。那么 A 和 E 之间的耦合关系是______。



编　号	参　数	返 回 值
1	数据项	数据项
2	数据项	数据项
3	功能码	无
4	无	列表

A. 公共耦合　　B. 数据耦合　　C. 内容耦合　　D. 无耦合

5.4.4 同步练习参考答案

1. C　　2. B　　3. D　　4. B　　5. A

5.5 系统测试

5.5.1 考点辅导

5.5.1.1 系统测试与调试

1. 系统测试的意义和目的

系统测试是为了发现错误而执行程序的过程。或者说，软件测试是根据软件开发各阶段的规格说明和程序的内部结构而精心设计一批测试用例(即输入数据及其预期的输出结果)，并利用这些测试用例去运行程序，以发现程序错误的过程。

信息系统测试应包括软件测试、硬件测试和网络测试。系统测试是保证系统质量和可靠性的关键步骤，是对系统开发过程中的系统分析、系统设计和实施的最后复查。根据测试的概念和目的，在进行信息系统测试时应遵循以下基本原则。

(1) 应尽早并不断地进行测试。

(2) 测试工作应该避免由原开发软件的人或小组承担。

(3) 设计测试方案的时候，不仅要确定输入数据，而且要根据系统功能确定预期的输出结果。

(4) 在设计用例时，不仅要设计有效、合理的输入条件，也要包含不合理、失效的输入条件。

(5) 在测试程序时，不仅要检验程序是否做了该做的事，还要检验程序是否做了不该做的事。

(6) 严格按照测试计划来进行，避免测试的随意性。

(7) 妥善保存测试计划、测试用例，作为软件文档的组成部分，为维护提供方便。

(8) 测试用例都是精心设计出来的，可以为重新测试或追加测试提供方便。

2. 测试过程

测试是开发过程中一个独立且非常重要的阶段，测试过程基本上与开发过程平行进行。一个规范的测试过程通常包括制订测试计划、编制测试大纲、根据测试大纲设计和生成测试用例、实施测试和生成测试报告。

5.5.1.2 测试策略和测试方法

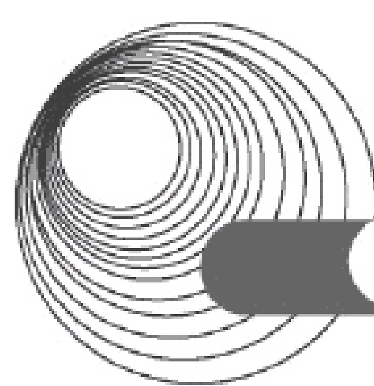
1. 软件测试策略

1) 单元测试

单元测试也称为模块测试。在模块编写完成且无法编译错误后就可以进行。如果选择机器测试，一般用白盒法，多个模块一起进行。

单元测试主要检查模块的以下 5 个特征。

- 模块接口。
- 局部数据结构。



- 重要的执行路径。
- 出错处理。
- 边界条件。

2) 组装测试

组装测试也称为集成测试，就是把模块按系统设计说明书的要求组合起来进行测试。组装测试有两种方法：一种是分别测试各个模块，再把这些模块组合起来进行整体测试，即非增量式集成；另一种是把下一个要测试的模块组合到已测试好的模块中，测试完后再将下一个需要测试的模块组合起来进行测试，逐步把所有模块组合在一起并完成测试，即增量式集成。

3) 确认测试

确认测试的任务是进一步检查软件的功能和性能是否与用户要求的一样。首先进行有效性测试以及软件配置审查，然后进行验收测试和安装测试，经过管理部门的认可和专家的鉴定后，软件即可交给用户使用。

4) 系统测试

系统测试是将已经确认的软件、计算机硬件、外设和网络等其他因素结合在一起，进行信息系统的各种组装测试和确认测试，其目的是通过与系统的需求相比较，发现所开发的系统与用户需求不符或矛盾的地方。常见的系统测试主要有恢复测试、安全性测试、强度测试、性能测试、可靠性测试和安装测试。

2. 测试方法

软件测试方法分为静态测试和动态测试。

1) 静态测试

静态测试是指被测试程序不在机器上运行，而是采用人工检测和计算机辅助静态分析的手段对程序进行检测。

(1) 人工检测。人工检测是不依靠计算机而是靠人工审查程序或评审软件，包括代码检查、静态结构分析和代码质量度量等。

(2) 计算机辅助静态分析。利用静态分析工具对被测试程序进行特性分析，从程序中提取一些信息，以便检查程序逻辑的各种缺陷和可疑的程序构造。

2) 动态测试

动态测试是指通过运行程序发现错误。对软件产品进行动态测试时可以采用黑盒测试法和白盒测试法。

测试用例的设计如下。

测试用例由测试输入数据和与之对应的预期输出结构组成。在设计测试用例时，应当包括合理的输入条件和不合理的输入条件。

(1) 用黑盒法设计测试用例。黑盒测试也称为功能测试，在完全不考虑软件的内部结构和特性的情况下，测试软件的外部特性。

常用的黑盒测试技术有等价类划分、边界值分析、错误推测和因果图等。

① 等价类划分。等价类划分法将程序的输入域划分为若干等价类，然后从每个等价类中选取一个代表性数据作为测试用例。每一类的代表性数据在测试中的作用等价于这一类中的其他值。这样就可以用少量具代表性的测试用例取得较好的测试效果。等价类划

分分两种不同的情况，即有效等价类和无效等价类。在设计测试用例时，要同时考虑这两种等价类。

② 边界值分析。输入的边界比中间更加容易发生错误，因此用边界值分析来补充等价类划分的测试用例设计技术。边界值分析选择等价类边界的测试用例，既注重于输入条件边界，又适用于输出域测试用例。

③ 错误推测。错误推测是基于经验和直觉推测程序中所有可能存在的错误，从而有针对性地设计测试用例的方法。其基本思想是列举出程序中所有可能有的错误和容易发生错误的特殊情况，根据它们选择测试用例。

④ 因果图。因果图法是从自然语言描述的程序规格说明中找出因(输入条件)和果(输出或程序状态的改变)，通过因果图转换为判定表。

(2) 用白盒法设计测试用例。白盒测试也称为结构测试，根据程序的内部结构和逻辑来设计测试用例，对程序的路径和过程进行测试，检查是否满足设计的需要。

白盒测试常用的技术是逻辑覆盖、循环覆盖和基本路径测试。

① 逻辑覆盖。逻辑覆盖考查用测试数据运行被测程序时对程序逻辑的覆盖程度。主要的逻辑覆盖标准有语句覆盖、判定覆盖、条件覆盖、判定/条件覆盖、条件组合覆盖和路径覆盖 6 种。

② 循环覆盖。执行足够的测试用例，使得循环中的每个条件都得到验证。

③ 基本路径测试。基本路径测试法是在程序控制流图的基础上，通过分析控制流图的环路复杂性，导出基本可执行路径集合，从而设计测试用例。

5.5.1.3　测试面向对象软件

对于面向对象软件，测试的基本目标仍然是在现实的时间范围内利用可控的工作量找出尽可能多的错误，但是其本质特征的不同使得测试策略和技术也发生了变化。

1. 单元测试

面向对象软件中单元的概念发生了变化，封装导出了类的定义。每个类和类的实例(对象)有属性(数据)和处理这些数据的操作(函数或方法)。封装的类常是单元测试的重点，然而，类中包含的操作是最小的可测试单元。由于类中可以包含一些不同的操作，且特殊的操作可以作为不同类的一部分存在，因此，面向对象软件的类测试是由封装在该类中的操作和类的状态行为驱动的。

2. 集成测试

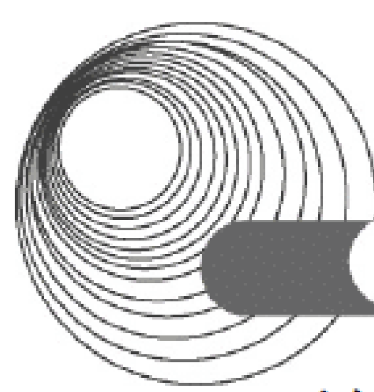
由于面向对象软件没有明显的层次控制结构，因此面向对象环境中的集成测试有两种策略：

(1) 基于线程的测试，对响应系统的一个输入或事件所需的一组类进行集成，每个线程单独地集成和测试，并应用回归测试以确保没有产生副作用。

(2) 基于使用的测试，通过测试很少使用服务类的那些类开始系统的构建。

5.5.1.4　测试 Web 应用

由于 WebApp 位于网络上，并与很多不同的操作系统、浏览器(位于很多不同的设备上)、硬件平台、通信协议及“暗中的”应用系统进行交互作用，因此，查找错误的工作是很重



要的。

为了了解 Web 工程环境中的测试目标，必须考虑 WebApp 质量的多种维度。

1. 质量维度

良好的设计应该将质量集成到 Web 应用中，通过对设计模型中的不同元素进行一系列技术评审，对质量进行评估。评估和测试都要检查下面质量维度中的一项或多项。

(1) 内容。

(2) 功能。

(3) 结构。

(4) 可用性。

(5) 导航性。

(6) 性能。

(7) 兼容性。

(8) 安全性。

2，WebApp 测试策略

WebApp 测试策略采用所有软件测试使用的基本原理，并建议使用面向对象系统使用的策略和战术。下面的步骤对此方法进行了总结。

(1) 对 WebApp 的内容模型进行评审，以发现错误。

(2) 对接口模型进行评审，保证适合所有的用例。

(3) 评审 WebApp 的设计模型，发现导航错误。

(4) 测试用户界面，发现表现机制和(或)导航机制中的错误。

(5) 对功能构件进行单元测试。

(6) 对贯穿体系结构的导航进行测试。

(7) 在各种不同的环境配置下实现 WebApp，并测试 WebApp 对于每一种配置的兼容性。

(8) 进行安全性测试，试图攻击 WebApp 或其所处环境的弱点。

(9) 进行性能测试。

(10) 通过可监控的最终用户群对 WebApp 进行测试，对他们与系统的交互结果进行以下方面的评估，包括内容和导航错误、可用性、兼容性以及 WebApp 的安全性、可靠性及性能等方面的评估。

5.5.1.5 测试方法

在软件测试过程中，应该为定义软件测试模板，即将特定的测试方法和测试用例设计放在一系列的测试步骤中。

软件测试方法分为静态测试和动态测试。

(1) 静态测试。静态测试是指被测试程序不在机器上运行，而是采用人工检测和计算机辅助静态分析的手段对程序进行检测。

① 人工检测。人工检测不依靠计算机而是依靠人工审查程序或评审软件，包括代码检查、静态结构分析和代码质量度量等。

② 计算机辅助静态分析。利用静态分析工具对被测试程序进行特性分析，从程序中提取一些信息，以便检查程序逻辑的各种缺陷和可疑的程序构造。

(2) 动态测试。动态测试是指通过运行程序发现错误。在对软件产品进行动态测试时可以采用黑盒测试法和白盒测试法。

测试用例由测试输入数据和与之对应的预期输出结果组成。在设计测试用例时，应当包括合理的输入条件和不合理的输入条件。

1. 黑盒测试

黑盒测试也称为功能测试，在完全不考虑软件的内部结构和特性的情况下，测试软件的外部特性。

2. 白盒测试

白盒测试也称为结构测试，根据程序的内部结构和逻辑来设计测试用例，对程序的路径和过程进行测试，检查是否满足设计的需要。

5.5.1.6　调试

调试的任务就是根据测试时所发现的错误，找出原因和具体的位置，进行改正。调试主要由程序开发人员来进行，谁开发的程序就由谁来进行调试。常用的调试方法有试探法、回溯法、对分查找法、归纳法和演绎法。

5.5.2　典型例题分析

例 1　某项目为了修正一个错误而进行了修改。错误修改后，还需要进行 (19) 以发现这一修改是否会引起原本正确运行的代码出错。(2013 年上半年试题 19)

(19) A. 单眼测试　　B. 接收测试　　C. 安装测试　　D. 回归测试

解析：回归测试是指修改了旧代码后，重新进行测试以确认修改没有引入新的错误或导致其他代码产生错误。

答案：D

例 2　在设计测试用例时，应遵循______原则。(2013 年上半年试题 35)

(35) A. 仅确定测试用例的输入数据，无需考虑输出结果

B. 只需检验程序是否执行应有的功能，不需要考虑程序是否做了多余的功能

C. 不仅要设计有效合理的输入，也要包含不合理、失效的输入

D. 测试用例应设计得尽可能复杂

解析：测试用例要包括欲测试的功能、应输入的数据和预期的输出结果。测试数据应该选用少量、高效的测试数据进行尽可能完备的测试。基本目标是：设计一组发现某个错误或某类错误的测试数据。测试用例应覆盖有效合理的输入，也要包括不合理、失效的输入。

答案：C

例 3　单元测试中，检验模块接口时，不需要考虑 (36) 。(2013 年上半年试题 36)

(36) A. 测试模块的输入参数和形式参数在个数、属性、单位上是否一致

B. 全局变量在各模块中的定义和用法是否一致

C. 输入是否改变了形式参数

D. 输入参数是否使用了尚未赋值或者尚未初始化的变量

解析：在单元测试的任务中，检查模块接口时需要考虑：输入的实际参数与形式参数是否一致；调用其他模块的实际参数与被调模块的形参是否一致；全程变量的定义在各模块中是否一致；外部输入、输出等。因此，D 选项不在考虑范围之内。

答案：D

例 4 若用白盒测试方法测试以下代码，并满足条件覆盖，则至少需要 (35) 个测试用例。采用 McCabe 度量法算出该程序的环路复杂性为 (36) 。(2015 年下半年试题 35、36)

```
int find _max(int i, int j, int k)
{
    int max;
    if(i>j) then
    if(i>k) then max =i;
    else max=k;
    else if(j>k) then max =j;
    else max=k;
}
```

(35) A. 3　　B. 4　　C. 5　　D. 6

(36) A. 1　　B. 2　　C. 3　　D. 4

解析：条件覆盖是指构造一组测试用例，使得每一判定语句中每个逻辑条件的各种可能的值至少满足一次。本题至少需要 4 组测试用例。

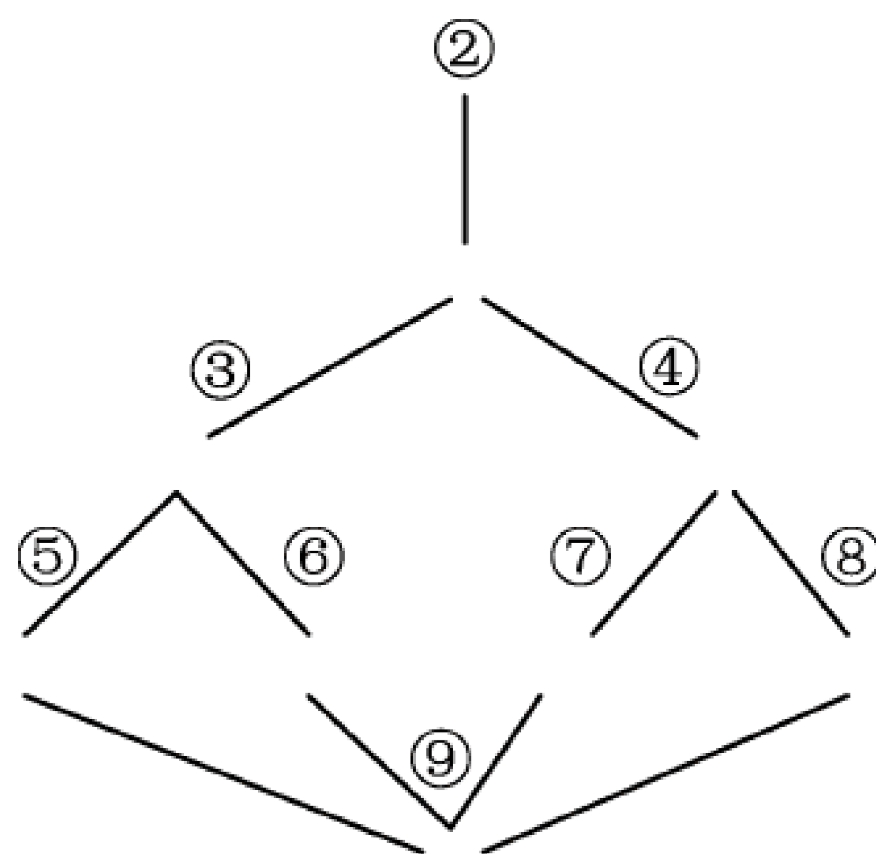
(1) $i>j$ 满足，$i>k$ 满足，如 $i=10, j=2, k=8$；

(2) $i>j$ 满足，$i>k$ 不满足，如 $i=10, j=2, k=13$；

(3) $i>j$ 不满足，$i>k$ 满足，如 $i=10, j=15, k=8$；

(4) $i>j$ 不满足，$i>k$ 不满足，如 $i=10, j=15, k=13$。

要计算 McCabe 复杂度需要先绘制出图，即



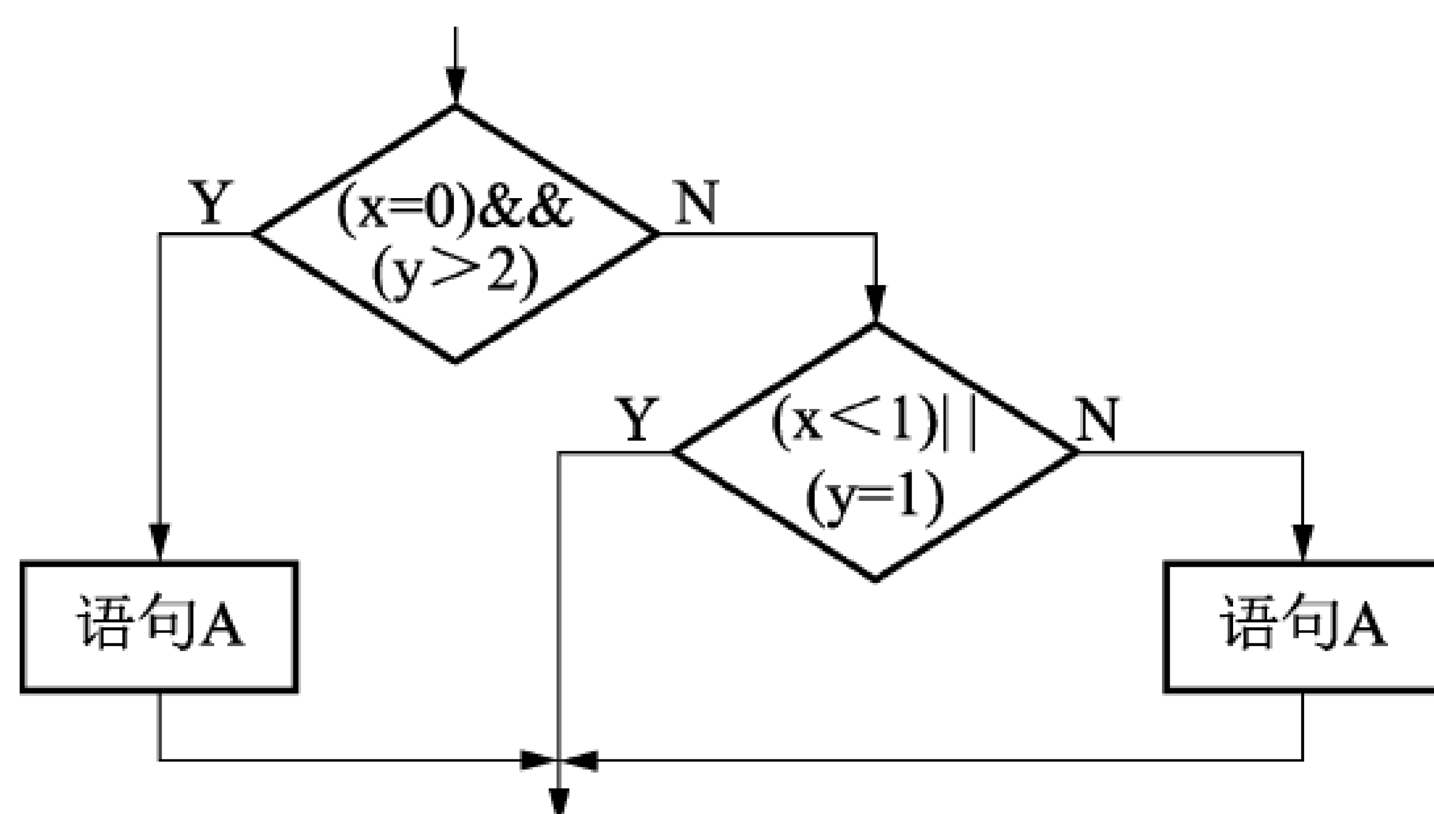
图中节点数为 9，弧度数为 11，强连通分量为 1，因此 McCabe 复杂度为：11−9+2=4。

答案：(35) B　　(36) D

例 5 采用白盒测试方法对下图进行测试，设计了 4 个测试用例：①(x=0，y=3)，②(x=l，y=2)，③(x=−1，y=2)，④(x=3, y=l)。至少需要测试用例①②才能完成 (35) 覆盖，至少需要测试用例①②③或①②④才能完成 (36) 覆盖。(2014 年上半年试题 35)

(35) A. 语句　　B. 条件　　C. 判定/条件　　D. 路径

(36) A. 语句　　B. 条件　　C. 判定/条件　　D. 路径



解析：语句覆盖要求被测程序中的每一条语句至少执行一次，这种覆盖对程序执行逻辑的覆盖很低。

条件覆盖要求每一判定语句中每个逻辑条件的各种可能的值至少满足一次。判定/条件覆盖要求判定中每个条件的所有可能取值(真/假)至少出现一次，并使得每个判定本身的判定结果(真/假)也至少出现一次。路径覆盖则要求覆盖被测程序中所有可能的路径。

通过测试用例①(x=0，y=3)，能执行到语句A，同时覆盖左侧路径；通过测试用例②(x=l, y=2)，能执行到语句B，同时覆盖右侧路径；通过测试用例③(x=-1，y=2)或④(x=3, y=l)，什么也不执行，覆盖中间路径。

答案：(35) A　(36) D

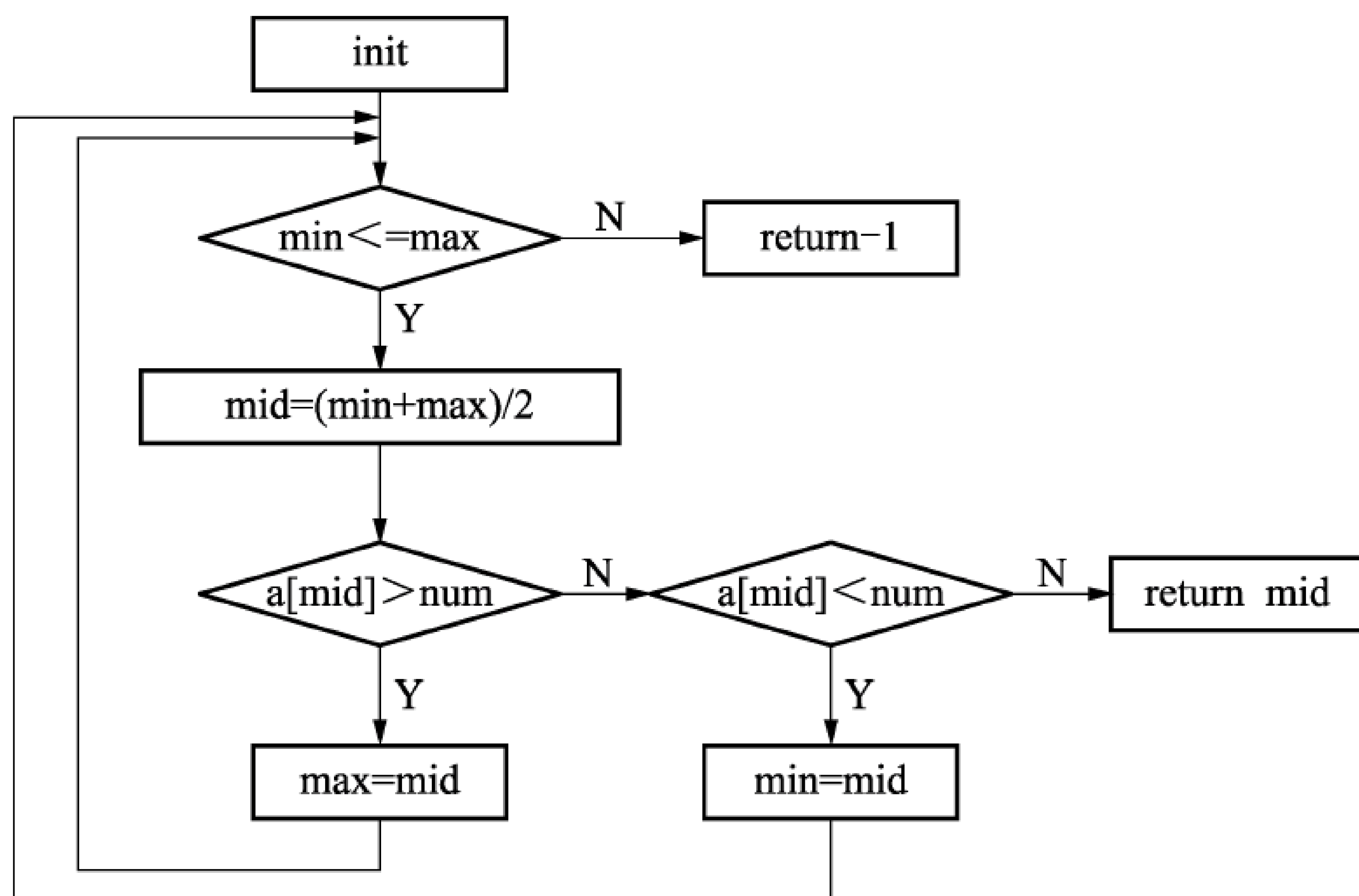
例 6　软件<u>(32)</u>是指一个系统在给定时间间隔内和给定条件下无失效运行的概率。(2011 年下半年试题 32)

(32) A．可靠性　　B．可用性　　C．可维护性　　D．可伸缩性

解析：软件可靠性是指在规定的条件下和规定的时间内，软件不引起系统故障的能力。可用性是指系统在特定使用环境下为特定用户用于特定用途时，所具有的有效性、效率和用户主观满意度。可维护性是指软件能够被理解、校正、适应及增强功能的容易程度。可伸缩性主要是指系统通过增加或减少服务器，从而提升或降低系统性能的难易程度。

答案：A

例 7　下图所示的逻辑流实现折半查找功能，最少需要<u>(34)</u>个测试用例可以覆盖所有的可能路径。(2011 年下半年试题 34)



(34) A. 1　　B. 2　　C. 3　　D. 4

解析：对于折半查找，有两种结果，要么查找到，要么没有查找到，这两种结果是互斥的，因此至少需要两个测试用例可以覆盖所有的可能路径。

答案：B

例 8　在某班级管理系统中，班级的班委有班长、副班长、学习委员和生活委员，且学生年龄在 15～25 岁。若用等价类划分来进行相关测试，则__(35)__不是好的测试用例。(2011 年下半年试题 35)

(35) A. (队长，15)　　B. (班长，20)　　C. (班长，15)　　D. (队长，12)

解析：等价类划分法选择适当的数据子集来代表整个数据集，通过降低测试的数目实现“合理的”覆盖，覆盖了更多的可能数据，以发现更多的软件缺陷。在设计测试用例的时候要考虑有效等价类和无效等价类。

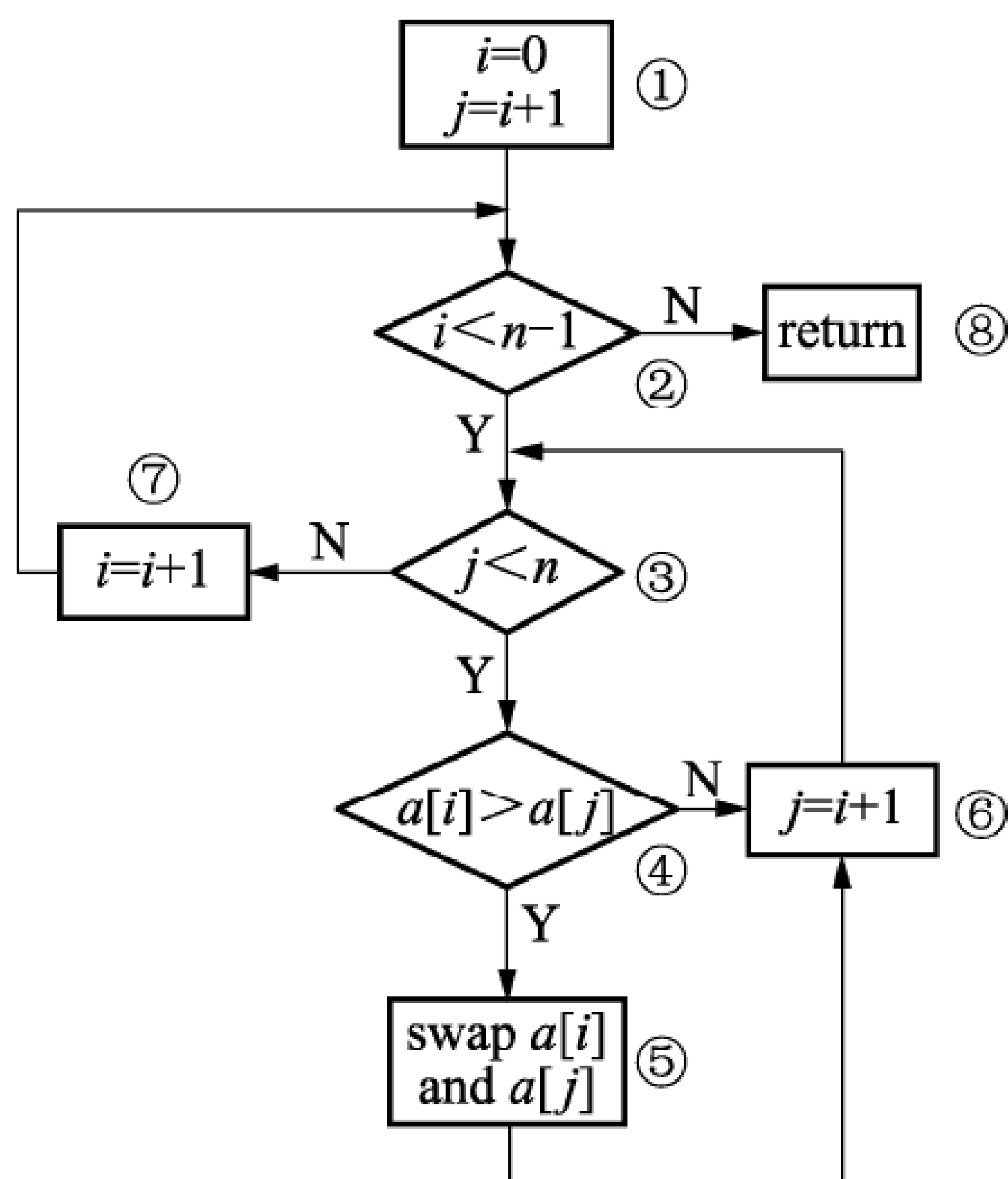
有效等价类：是指输入完全满足程序输入的规格说明，有效、有意义的输入数据所构成的集合。利用有效等价类可以检验程序是否满足规格说明所规定的功能和性能。

无效等价类：和有效等价类相反，即不满足程序输入要求或者无效的输入数据构成的集合。使用无效等价类，可以鉴别程序异常情况的处理。

本题中，(队长，15)、(队长，12)都是无效等价类的测试用例，当测试到队长不属于班委干部时就会抛出异常，对年龄的测试就没有意义了，因此(队长，12)不是好的测试用例。可以设计(班长，12)测试用例，用于测试年龄不符合要求的情况。

答案：D

例 9　下图所示的逻辑流，最少需要__(35)__个测试用例可实现语句覆盖。(2011 年上半年试题 35)



(35) A. 1　　B. 2　　C. 3　　D. 5

解析：语句覆盖是指选择足够的测试数据，使被测试程序中的每条语句至少执行一次。本题是冒泡排序算法的逻辑流，最少需要一个测试用例可实现语句覆盖，用例数组可以选为 a={3,4,2,5,1}。

答案：A

例 10　在软件开发过程中，系统测试阶段的测试目标来自于__(32)__阶段。(2014 年下半年试题 32)

(32) A．需求分析　　B．概要设计　　C．详细设计　　D．软件实现

解析：系统测试是针对整个产品系统进行的测试，目的是验证系统是否满足了需求规格的定义，找出与需求规格不符或与之矛盾的地方，从而提出更加完善的方案。

答案：A

例 11　自底向上的集成测试策略的优点包括__(34)__。(2015 年上半年试题 34)

(34) A. 主要的设计问题可以在测试早期处理
B. 不需要写驱动程序
C. 不需要写桩程序
D. 不需要进行回归测试

解析：自底向上集成方式从程序模块结构中最底层的模块开始组装和测试。因为模块常常是自底向上进行组装的，对于一个给定层次的模块来说，它的子模块及下属模块已经组装并测试完成，所以不再需要桩模块。

优点：对底层组件行为较早验证；工作最初可以并行集成，比自顶向下效率高；减少了桩的工作量；支持故障隔离。

缺点：驱动的开发工作量大；对高层的验证被推迟，设计上的错误不能被及时发现。

答案：C

例 12　不属于黑盒测试技术的是__(36)__。(2010 年下半年试题 36)

(36) A．错误猜测　　B．逻辑覆盖　　C．边界值分析　D．等价类划分

解析：黑盒测试也称为功能测试，在完全不考虑软件内部结构和特性的情况下，测试软件的外部特性。常用的黑盒测试技术有等价类划分、边界值分析、错误推测和因果图等。逻辑覆盖是白盒测试中用到的方法。

答案：B

例 13　以下关于可视化程序设计的叙述中，错误的是__(49)__。(2010 年下半年试题 49)

(49) A．可视化程序设计使开发应用程序无需编写程序代码
B．可视化程序设计基于面向对象的思想，引入了控件和事件驱动
C．在可视化程序设计中，构造应用程序界面就像搭积木
D．在可视化程序设计中，采用解释方式可随时查看程序的运行效果

解析：可视化(Visual)程序设计是一种全新的程序设计方法，它主要是让程序设计人员利用软件本身所提供的各种控件，像搭积木似地构造应用程序的各种界面。

可视化程序设计以“所见即所得”的编程思想为原则，力图实现编程工作的可视化，即随时可以看到结果，程序与结果的调整同步。

可视化编程语言的特点主要表现在两个方面：一是基于面向对象的思想，引入了控件的概念和事件驱动；二是程序开发过程一般遵循以下步骤，即先进行界面的绘制工作，再基于事件编写程序代码，以响应鼠标、键盘的各种动作。

可视化程序设计最大的优点是设计人员可以不用编写或只需编写很少的程序代码，就能完成应用程序的设计，这样就能极大地提高设计人员的工作效率。

答案：A

例 14 使用白盒测试方法时，应根据 (17) 和指定的覆盖标准确定测试数据。(2010年上半年试题17)

(17) A．程序的内部逻辑　　B．程序结构的复杂性

C．使用说明书　　D．程序的功能

解析：白盒测试也称为结构测试，根据程序的内部结构和逻辑来设计测试用例，对程序的执行路径和过程进行测试，检查是否满足设计的需要。白盒测试常用的技术涉及不同的覆盖标准，在测试时需根据制定的覆盖标准确定测试数据。

答案：A

例 15 一个功能模块 M_1 中的函数 F_1 有一个参数需要接收指向整型的指针，但是在功能模块 M_2 中调用 F_1 时传递了一个整型值，在软件测试中， (35) 最可能测出这一问题。(2010年上半年试题35)

(35) A．M_1 的单元测试　　B．M_2 的单元测试

C．M_1 和 M_2 的集成测试　　D．确认测试

解析：单元测试侧重于模块中的内部处理逻辑和数据结构，所有模块都通过了测试之后，把模块集成起来仍可能会出现穿越模块的数据丢失、模块之间的相互影响等问题，因此，需要模块按系统设计说明书的要求组合起来进行测试，即集成测试，以发现模块之间协作的问题。

一个功能模块 M_1 中的函数 F_1 有一个参数需要接收指向整型的指针，但是在功能模块 M_2 中调用 F_1 时传递了一个整型值，这种模块之间传递参数的错误，在集成测试中最可能测试出来。

答案：C

例 16 一个项目为了修正一个错误而进行了变更。但这个错误被修正后，却引起以前可以正确运行的代码出错。 (18) 最可能发现这一问题。(2009年下半年试题18)

(18) A．单元测试　　B．接受测试　　C．回归测试　　D．安装测试

解析：单元测试也称为模块测试，在模块编写完成且无编译错误后就可以进行。单元测试侧重于模块中的内部处理逻辑和数据结构。

接受测试是经过集成测试之后，软件被集成起来，接口方面的问题已经解决，将进入软件测试的最后一个环节，即确认测试。确认测试的任务是进一步检查软件的功能和性能是否与用户的要求一样。

在软件生命周期中的任何一个阶段，只要软件发生了改变，就可能给该软件带来问题。软件的改变可能是源于发现了错误并做了修改，也有可能是因为在集成或维护阶段加入了新的模块。当软件中所含错误被发现时，如果错误跟踪与管理系统不够完善，就可能会遗漏对这些错误的修改；而开发者对错误理解得不够透彻，也可能导致所做的修改只修正了错误的外在表现，而没有修复错误本身，从而造成修改失败；修改还有可能产生副作用，从而导致软件未被修改的部分产生新的问题，使本来工作正常的功能产生错误。同样，在有新代码加入软件的时候，除了新加入的代码中有可能含有错误外，新代码还有可能对原有的代码带来影响。因此，每当软件发生变化时，必须重新测试现有的功能，以便确定修改是否达到了预期的目的，检查修改是否损害了原有的正常功能。同时，还需要补充新的测试用例来测试新的或被修改了的功能。为了验证修改的正确性及其影响就需要进行回归测试。

有了测试用例库的维护方法和回归测试包的选择策略，回归测试可遵循下述基本过程进行。

(1) 识别出软件中被修改的部分。

(2) 从原基线测试用例库 T 中，排除所有不再适用的测试用例，确定那些对新的软件版本依然有效的测试用例，其结果是建立一个新的基线测试用例库 T_0。

(3) 依据一定的策略从 T_0 中选择测试用例测试被修改的软件。

(4) 如果必要，生成新的测试用例集 T_1，用于测试 T_0 无法充分测试的软件部分。

(5) 用 T_1 执行修改后的软件。

第(2)步和第(3)步测试验证修改是否破坏了现有的功能，第(4)步和第(5)步测试验证修改工作本身。

安装测试就是为了检测在安装过程中是否有误、是否容易操作等。

答案：C

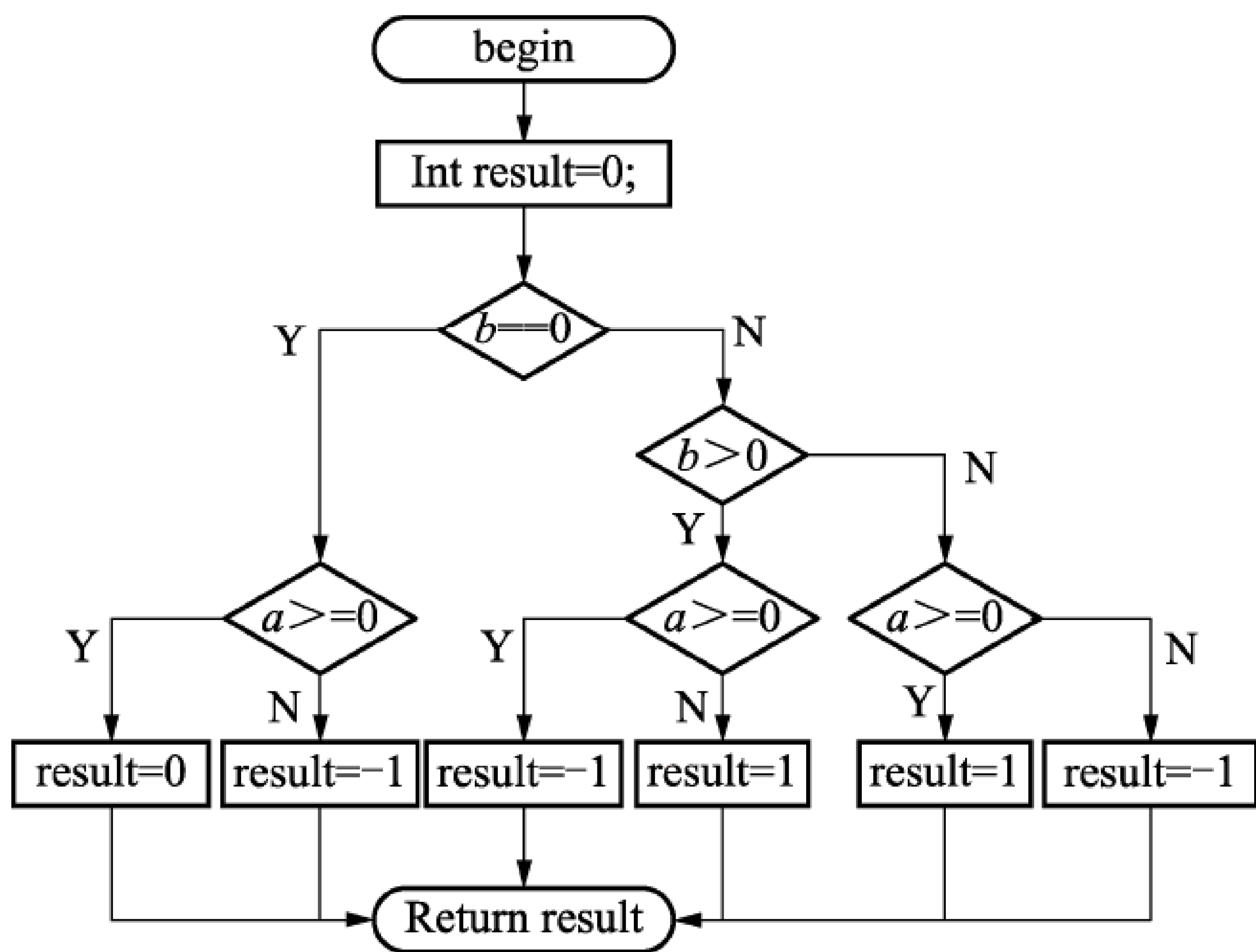
例 17　某系统重用了第三方组件(但无法获得其源代码)，则应采用 __(35)__ 对组件进行测试。(2009 年下半年试题 35)

(35) A．基本路径覆盖　　B．分支覆盖　　C．环路覆盖　　D．黑盒测试

解析：因为本题中重用的第三方组件无法获知其源代码，因此属于白盒测试的基本路径覆盖、分支覆盖和环路覆盖都不适用，而黑盒测试是在完全不考虑软件的内部结构和特性的情况下，测试软件的外部特性，因此可以达到测试的目的。

答案：D

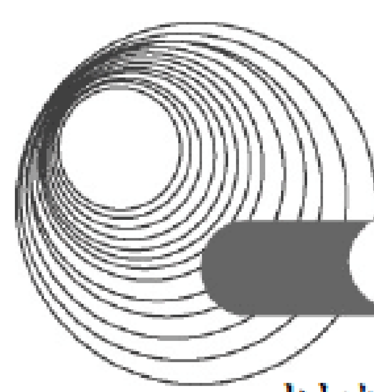
例 18　当用分支覆盖法对下图所示的流程图进行测试时，至少需要设计 __(35)__ 个测试用例。(2009 年上半年试题 35)



(35) A．4　　B．5　　C．6　　D．8

解析：分支覆盖属于白盒测试中的逻辑覆盖，分支覆盖就是设计若干测试用例，运行所测程序，使得程序中每个判断的取真分支和取假分支至少经历一次。

令第一层的 b=0 判断节点的左分支为 1，右分支为 2；第二层的 b>0 判断节点的左分支为 3，右分支为 4；第三层左边 a≥0 判断节点的左分支为 5，右分支为 6；第三层中间 a≥0



判断节点的左分支为 7，右分支为 8；第三层右边 $a \geq 0$ 判断节点的左分支为 9，右分支为 10；则进行分支覆盖的测试路径为 1、5，1、6，2、3、7，2、3、8，2、4、9，2、4、10。共 6 个测试用例。

答案：C

5.5.3 同步练习

1. 某程序根据输入的 3 条线段长度，判断这 3 条线段能否构成三角形。以下 6 个测试用例中，______两个用例属于同一个等价类。

① 6、7、13　② 4、7、10　③ 9、20、35

④ 9、11、21　⑤ 5、5、4　⑥ 4、4、4

A. ①②　B. ③④　C. ⑤⑥　D. ①④

2. 在模拟环境下，常采用黑盒测试检验所开发的软件是否与需求规格说明书一致。其中有效性测试属于______中的一个步骤。

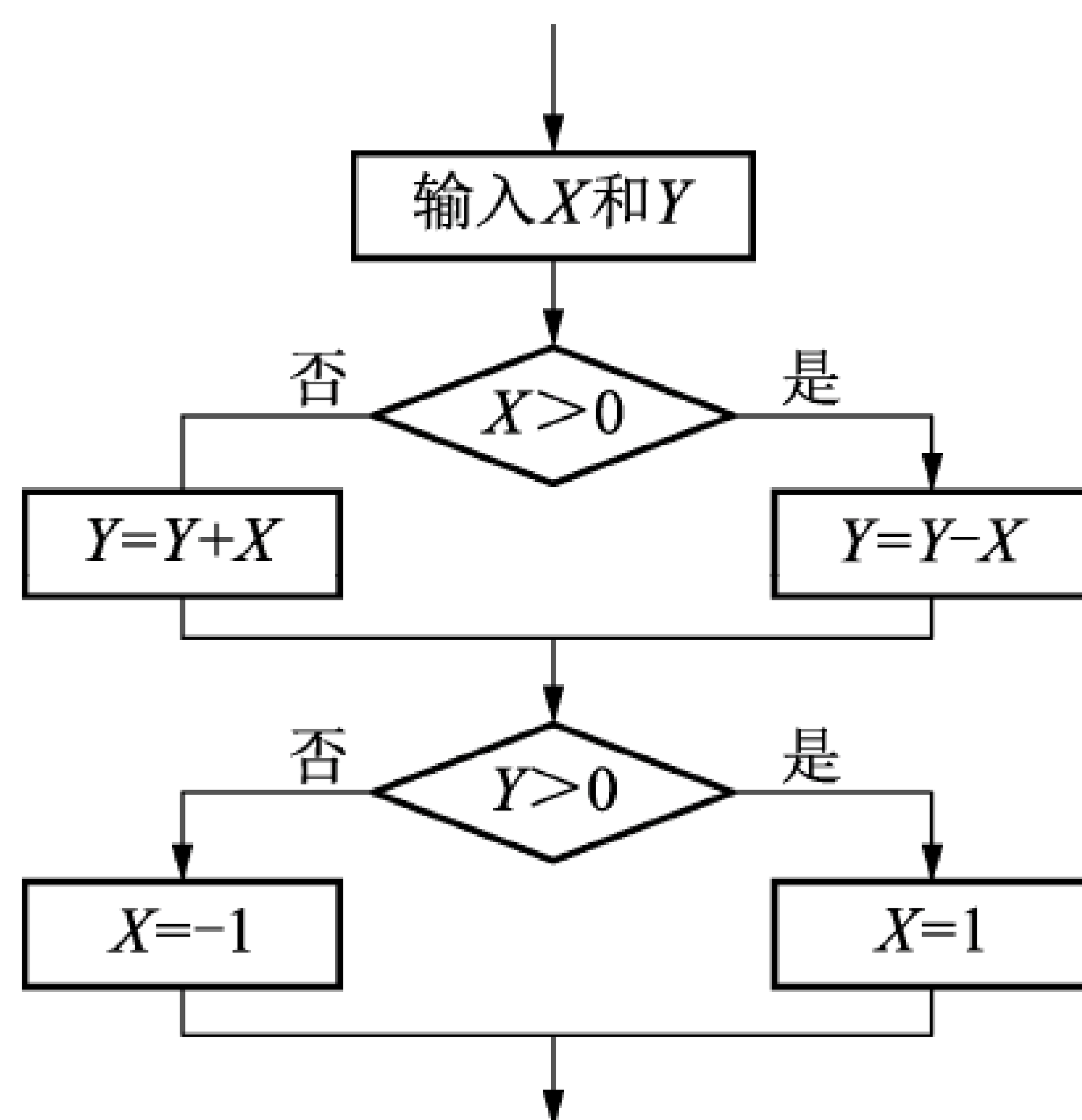
A. 单元测试　B. 集成测试　C. 确认测试　D. 系统测试

3. 软件测试是软件开发中不可缺少的活动，通常 (1) 在代码编写阶段进行。检查软件的功能是否与用户要求一致是 (2) 的任务。

(1)～(2) A. 验收测试　B. 系统测试

C. 单元测试　D. 集成测试

4. 对于下图所示的程序流程，当采用语句覆盖法设计测试案例时，至少需要设计个测试案例。



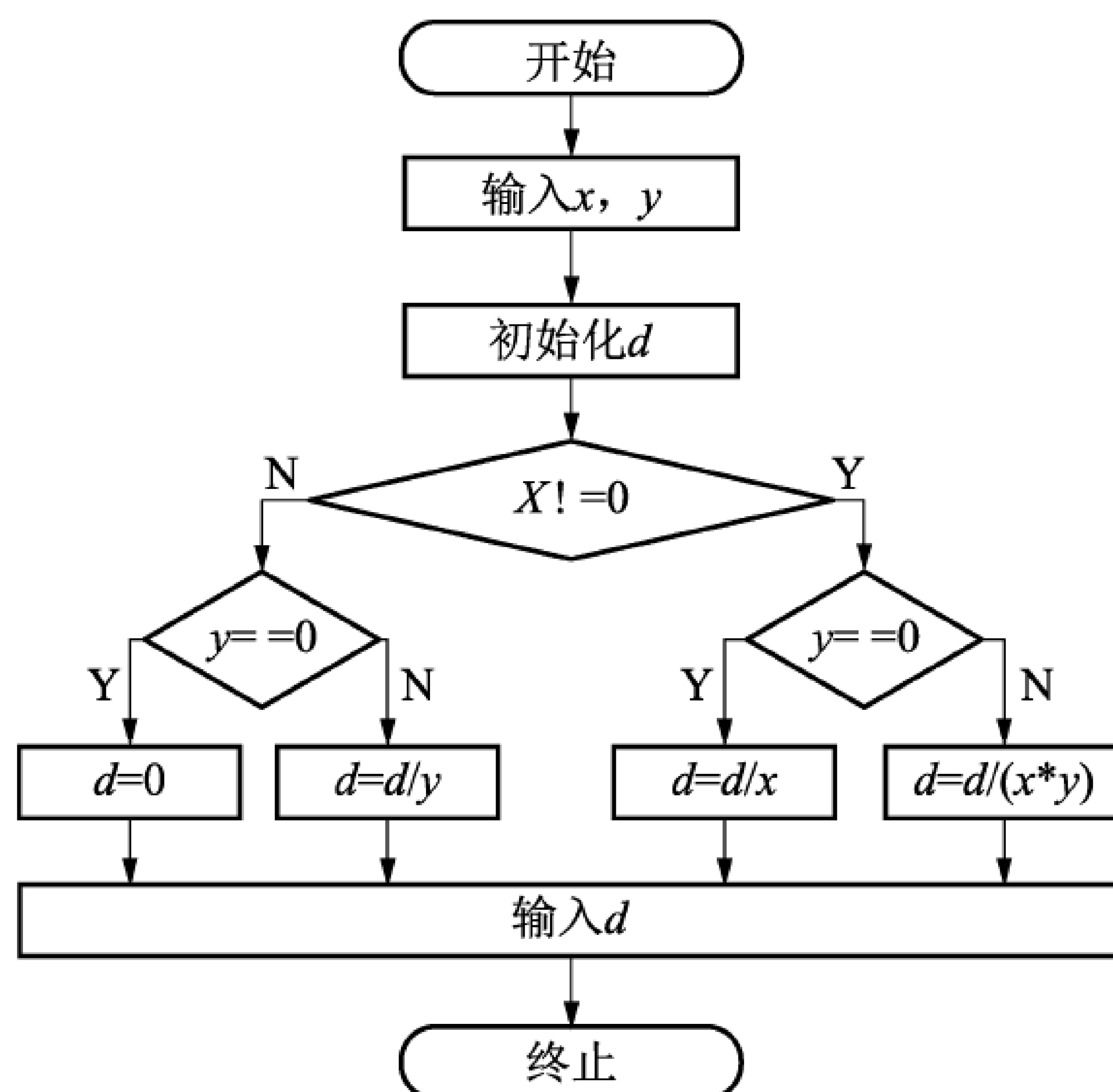
A. 1　B. 2　C. 3　D. 4

5. 为验证程序模块 A 是否正确实现了规定的功能，需要进行 (1) ；为验证模块 A 能否与其他模块按照规定方式正确工作，需要进行 (2) 。

(1)～(2) A. 单元测试　B. 集成测试

C. 确认测试　D. 系统测试

6. 阅读下图所示的流程图。



当用判定覆盖法进行测试时，至少需要设计______个测试用例。

A. 2　　B. 4　　C. 6　　D. 8

7. 在某大学学生学籍管理信息系统中，假设学生年龄的输入范围为 16～40，则根据黑盒测试中的等价类划分技术，下面划分正确的是______。

A. 可划分为 2 个有效等价类，2 个无效等价类
B. 可划分为 1 个有效等价类，2 个无效等价类
C. 可划分为 2 个有效等价类，1 个无效等价类
D. 可划分为 1 个有效等价类，1 个无效等价类

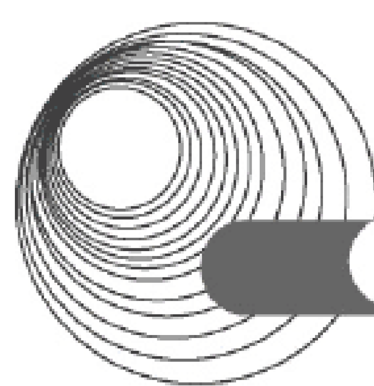
8. 给定 C 语言程序。

```
int foo( int x,  int y,  int d)
{
  if( x != 0 )
  {
  if ( y == 0 ) d = d / x;
  else d = d / (x * y );
  }
  else
  {
  if( y == 0 ) d = 0;
  else d = d / y;
   }
   return d ;
}
```

当用路径覆盖法进行测试时，至少需要设计_____个测试用例。

A. 3　　B. 4　　C. 5　　D. 8

9. 软件的测试通常分为单元测试、组装测试、确认测试、系统测试 4 个阶段进行。_____属于确认测试阶段的活动。



A．设计评审　　B．代码审查　　C．结构测试　　D．可靠性测试

10．两个小组独立地测试同一个程序，第一组发现 25 个错误，第二组发现 30 个错误，在两个小组发现的错误中有 15 个是共同的，那么可以估计程序中的错误总数是_____个。

A．25　　B．30　　C．50　　D．60

11．下面有关测试的说法，正确的是______。

A．测试人员应该在软件开发结束后开始介入

B．测试主要是软件开发人员的工作

C．要根据软件详细设计中设计的各种合理数据设计测试用例

D．严格按照测试计划进行，避免测试的随意性

12．在系统转换的过程中，旧系统和新系统并行工作一段时间，再由新系统代替旧系统的策略称为 (1) ；在新系统全部正式运行前，一部分一部分地代替旧系统的策略称为 (2) 。

(1)～(2) A．直接转换　　B．位置转换

C．分段转换　　D．并行转换

5.5.4　同步练习参考答案

1．B	2．C
3．(1) C　(2) A	4．B
5．(1) A　(2) B	6．B
7．B	8．B
9．D	10．C
11．D	12．(1) D　(2) C

5.6　运行和维护知识

5.6.1　考点辅导

5.6.1.1　系统转换

在进行新、老系统转换以前，首先要进行新系统的试运行。系统试运行阶段的主要工作有：对系统进行初始化，输入各原始数据记录；记录系统运行的数据和状况；核对新系统输出和老系统输出的结果；对实际系统的输入方式进行考查；对系统实际运行、响应速度进行测试。

新系统运行成功之后，就可以在新系统和老系统之间互相转换。新、老系统之间的转换方式有直接转换、并行转换和分段转换。

(1) 直接转换。在确定新系统运行无误后，立刻启用新系统，终止老系统运行。

(2) 并行转换。新、老系统并行一段时间，经过一段时间的考验以后，新系统正式代替老系统。

(3) 分段转换。其又称逐步转换、向导转换、试点过渡法等。这种方法实际上是以上两种转换方式的结合。

5.6.1.2　系统维护概述

1. 系统可维护性的概念

系统可维护性的定义：维护人员理解、改正、改动和改进这个软件的难易程度。提高可维护性是开发管理信息系统所有步骤的关键，系统是否能够被很好地维护，可以用系统的可维护性这一指标来衡量。

(1) 系统的可维护性指标：可理解性、可测试性和可修改性。

(2) 维护与软件文档：文档是软件可维护性的决定因素。软件系统的文档可以分为用户文档和系统文档两类。

(3) 软件文档的修改：每当对数据、软件结构、模块过程或任何其他有关的软件特点有了改动时，必须立即修改相应的技术文档。

2. 系统维护的内容及类型

1) 硬件维护

硬件维护应由专职的硬件维护人员来负责，主要有两种类型的维护活动：一种是定期的设备保养性维护；另一种是突发性的故障维护。

2) 软件维护

软件维护主要是根据需求变化或硬件环境的变化对应用程序进行部分或全部的修改。软件维护的内容包括正确性维护、适应性维护、完善性维护和预防性维护等。

(1) 正确性维护。在软件交付使用后，必然会有一部分隐藏的错误被带到运行阶段。这些隐藏下来的错误在某些特定的使用环境下就会暴露出来。为了识别和纠正软件错误、改正软件性能上的缺陷、排除实施中的误使用，应当进行的诊断和改正错误的过程，就叫做正确性维护。

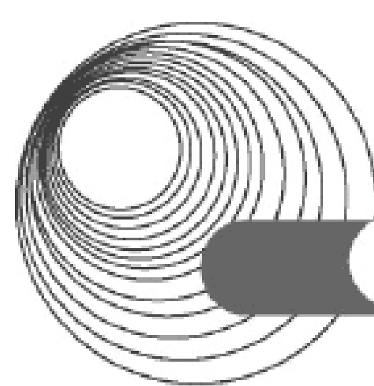
(2) 适应性维护。随着计算机的飞速发展，外部环境(新的硬、软件配置)或数据环境(数据库、数据格式、数据输入输出方式、数据存储介质)可能发生变化，为了使软件适应这种变化，而去修改软件的过程就叫做适应性维护。

(3) 完善性维护。在软件的使用过程中，用户往往会对软件提出新的功能与性能要求。为了满足这些要求，需要修改或再开发软件，以扩充软件功能、增强软件性能、改进加工效率、提高软件的可维护性。这种情况下进行的维护活动叫做完善性维护。

(4) 预防性维护。为了改进应用软件的可靠性和可维护性，为了适应未来的软、硬件环境的变化，应主动增加预防性的新功能，以使应用系统适应各类变化而不被淘汰。例如，将专用报表功能改成通用报表生成功能，以适应将来报表格式的变化。这方面的维护工作量占整个维护工作量的 4%左右。

3) 数据维护

数据维护主要是由数据库管理员来负责，主要负责数据库的安全性和完整性以及进行并发性控制。



3. 系统维护的管理和步骤

系统维护的管理和步骤如下。

(1) 提出维护或修改要求。

(2) 领导审查并作出答复，如同意修改则列入维护计划。

(3) 领导分配任务，维护人员修改。

(4) 验收维护成果并登记修改信息。

5.6.1.3 系统评价

1. 系统评价概述

信息系统的评价分为广义和狭义两种。广义的信息系统评价是指从系统开发的一开始到结束的每一阶段都需要进行评价。狭义的信息系统评价则是指在系统建成并投入运行之后所进行的全面、综合的评价。

按评价的时间与信息系统所处阶段的关系，又可从总体上把广义的信息系统评价分成立项评价、中期评价和结项评价。

2. 系统评价的指标

从以下几方面综合考虑，建立起一套指标体系理论框架。

(1) 从信息系统的组成部分出发，信息系统是一个由人机共同组成的系统，所以可以按照运行效果和用户需求(人)、系统质量和技术条件(机)这两条线索构造指标。

(2) 从信息系统的评价对象出发，对于开发方来说，他们所关心的是系统质量和技术水平；对于用户方而言，他们关心的是用户需求和运行质量；系统外部环境则主要通过社会效益指标来反映。

(3) 从经济学角度出发，分别按系统成本、系统效益和财务指标 3 条线索建立指标。

5.6.2 典型例题分析

例 1 在软件维护中，由于企业的外部市场环境和管理需求的变化而进行的维护工作，属于 (33) 维护。(2013 年上半年试题 33)

(33) A. 正确性 B. 适应性 C. 完善性 D. 预防性

解析：软件维护包括：①改正性维护，是指改正在系统开发阶段已发生而系统测试阶段尚未发现的错误；②适应性维护，是指使用软件适应信息技术变化和管理需求变化而进行的修改；③完善性维护，这是为扩充功能和改善性能而进行的修改，主要是指对已有的软件系统增加一些在系统分析和设计阶段中没有规定的功能与性能特征；④预防性维护是指为了改进应用软件的可靠性和可维护性，为了适应未来的软、硬件环境的变化，主动增加预防性的新功能，以使应用系统适应各类变化而不被淘汰。显然，该题中的情况属于适应性维护。

答案：B

例 2 由于信用卡公司升级了其信用卡支付系统，导致超市的原有信息系统也需要做相应的修改工作，该类维护属于 (34) 。(2012 年下半年试题 34)

(34) A．正确性维护　B．适应性维护　C．完善性维护　D．预防性维护

解析： 正确性维护是指改正在系统开发阶段已发生而系统测试阶段尚未发现的错误；适应性维护是指应用软件适应信息技术变化和管理需求变化而进行的修改；完善性维护是为扩充功能和改善性能而进行的修改，主要是指对已有的软件系统增加一些在系统分析和设计阶段没有规定的功能和性能特征。预防性维护是指为了改进应用软件的可靠性和可维护性，为了适应未来的软、硬件环境的变化，应主动增加预防性的新功能，以使应用系统适应各种变化而不被淘汰。本例的系统是适应信用卡支付系统的升级而作出的修改，属于适应性维护。

答案： B

例 3　某企业由于外部市场环境和管理需求的变化对现有软件系统提出新的需求，则对该软件进行的维护属于<u>(31)</u>维护。(2012 年上半年试题 31)

(31) A. 正确性　B. 完善性　C. 适应性　D. 预防性

解析： 适应性维护是指应用软件适应信息技术变化和管理需求变化而进行的修改。如改善系统硬件环境而产生系统更新换代的需求，企业由于外部市场环境和管理需求的变化对现有软件系统提出新的需求等。

正确性维护是指改正系统开发阶段已发生而系统测试阶段尚未发现的错误。

完善性维护主要是指对已有的软件系统增加一些在系统分析和设计阶段中都没有规定的功能和性能特征。

预防性维护则是为了适应未来的软硬件环境的变化，主动增加预防性的新功能，以使应用系统适应各类变化而不被淘汰。

答案： C

例 4　Mc Call 软件质量模型从软件产品的运行、修正和转移 3 个方面确定了 11 个质量特性，其中<u>(32)</u>不属于产品运行方面的质量特性。(2012 年上半年试题 32)

(32) A. 正确性　B. 可靠性　C. 效率　D. 灵活性

解析： 产品运行方面的质量特性有正确性、可靠性、易使用性、效率、完整性。灵活性则属于产品修正方面的特性。

答案： D

例 5　以下关于软件可维护性的叙述中，不正确的是“可维护性<u>(36)</u>”。(2015 年上半年试题 36)

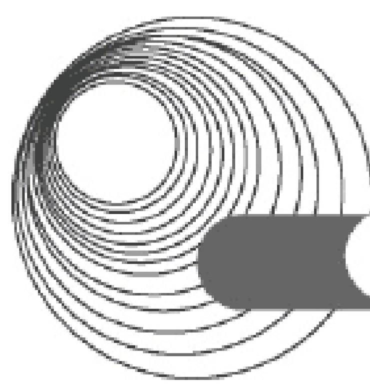
(36) A．是衡量软件质量的一个重要特性
B．不受软件开发文档的影响
C．是软件开发阶段各个时期的关键目标
D．可以从可理解性、可靠性、可测试性、可行性、可移植性等方面进行度量

解析： 文档是软件可维护性的决定因素。由于长期使用的大型软件在使用过程中必然会经受多次修改，所以文档就显得非常重要。

答案： B

例 6　针对应用在运行期的数据特点，修改其排序算法使其更高效，属于<u>(33)</u>维护。(2011 年下半年试题 33)

(33) A. 正确性　B. 适应性　C. 完善性　D. 预防性



解析：软件维护一般分为正确性维护、适应性维护、完善性维护和预防性维护。完善性维护是对软件功能的扩展和对性能的改善。题目中“修改其排序算法使其更高效”很明显是对性能的改善，属于完善性维护。

答案：C

例 7 进行防错性程序设计，可以有效地控制 (36) 维护成本。(2011 年下半年试题 36)

(36) A. 正确性 B. 适应性 C. 完善性 D. 预防性

解析：正确性维护是改正在系统开发阶段已发生而系统测试阶段尚未发现的错误。进行防错性程序设计，可以有效地控制正确性维护成本。

答案：A

例 8 高质量的文档所应具有的特性中，不包括 (33) 。(2011 年上半年试题 33)

(33) A．针对性，文档编制应该考虑读者对象群

B．精确性，文档的行文应当十分准确，不能出现多义性的描述

C．完整性，任何文档都应当是完整的、独立的，应该自成体系

D．无重复性，同一软件系统的几个文档之间应该没有相同的内容，若确实存在相同内容，则可以用“见**文档**节”的方式引用

解析：为使软件文档能起到多种桥梁的作用，使它有助于程序员编制程序，有助于管理人员监督和管理软件的开发，有助于用户了解软件的工作和应做的操作，有助于维护人员进行有效的修改和扩充，文档的编制必须保证一定的质量。

① 针对性。文档编制以前应分清读者对象。按不同的类型、不同层次的读者，决定怎样适应他们的需要。

② 精确性。文档的行文应当十分确切，不能出现多义性的描述。同一课题几个文档的内容应当是协调一致，没有矛盾的。

③ 清晰性。文档编写应力求简明，如有可能，配以适当的图表，以增强其清晰性。

④ 完整性。任何一个文档都应当是完整的、独立的，并应自成体系。同一课题的几个文档之间可能有些部分内容相同，这种重复是必要的。不要在文档中出现转引其他文档内容的情况。例如，一些段落没有具体描述，而用“见××文档××节”的方式，这将给读者带来许多不便。

⑤ 灵活性。各个不同软件项目，其规模和复杂程度有着许多实际差别，能一律看待。

答案：D

例 9 在软件维护阶段，为软件的运行增加监控设施属于 (34) 维护。(2011 年上半年试题 34)

(34) A．改正性 B．适应性 C．完善性 D．预防性

解析：完善性维护主要是指对已有的软件系统添加一些在系统分析和设计阶段中没有规定的功能与性能特征。为软件的运行增加监控设施是新增的功能，属于完善性维护。

答案：C

例 10 以下关于软件维护和可维护性的叙述中，不正确的是 (36) 。(2014 年下半年试题 36)

(36) A．软件维护要解决软件产品交付用户之后运行中发生的各种问题。

B．软件的维护期通常比开发期长得多，其投入也大得多

C．进行质量保证审查可以提高软件产品的可维护性。

D．提高可维护性是在软件维护阶段考虑的问题

解析：在软件设计阶段就应该考虑软件的可维护性。

答案：D

例 11　某银行为了使其网上银行系统能够支持信用卡多币种付款功能而进行扩充升级，这需要对数据类型稍微进行一些改变，这一状况需要对网上银行系统进行 (36) 维护。(2009 年上半年试题 36)

(36) A．正确性　　B．适应性　　C．完善性　　D．预防性

解析：正确性维护：在软件交付使用后，因开发时测试得不彻底、不完全，必然会有部分隐藏的错误遗留到运行阶段。这些隐藏下来的错误在某些特定的使用环境下就会暴露出来。为了识别和纠正软件错误、改正软件性能上的缺陷、排除实施中的误使用，应当进行的诊断和改正错误的过程就叫做正确性维护。

适应性维护：在使用过程中，外部环境(新的硬、软件配置)和数据环境(数据库、数据格式、数据输入输出方式、数据存储介质)可能发生变化。为使软件适应这种变化，而去修改软件的过程就叫做适应性维护。某银行为了使其网上银行系统能够支持信用卡多币种付款功能而进行扩充升级，因此选 B。

完善性维护：在软件的使用过程中，用户往往会对软件提出新的功能与性能要求。为了满足这些要求，需要修改或再开发软件，以扩充软件功能、增强软件性能、改进加工效率、提高软件的可维护性。这种情况下进行的维护活动叫做完善性维护。

预防性维护：预防性维护是为了提高软件的可维护性、可靠性等，为以后进一步改进软件打下良好基础。

答案：B

5.6.3　同步练习

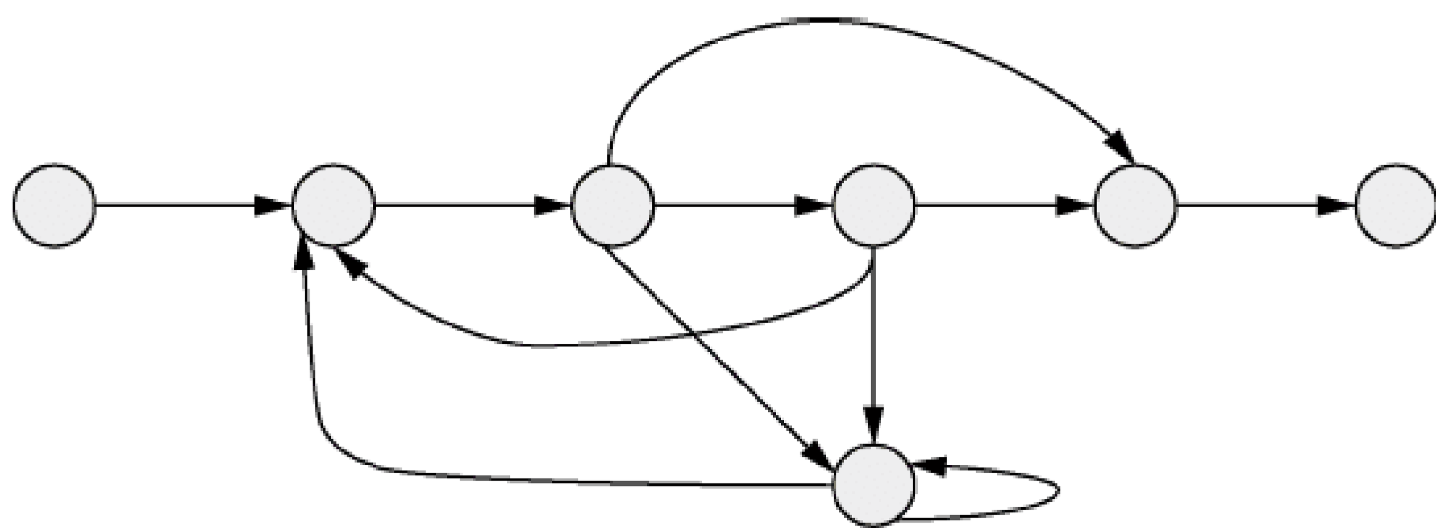
1.　统一过程(UP)定义了初启阶段、精化阶段、构建阶段、移交阶段和产生阶段，每个阶段以达到某个里程碑为结束，其中______的里程碑是生命周期架构。(2010 年上半年试题 32)

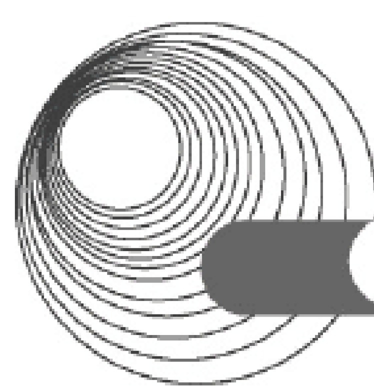
A．初启阶段　　B．精化阶段　　C．构建阶段　　D．移交阶段

2.　软件配置管理的内容不包括______。(2015 年上半年试题 31)

A．版本控制　　B．变更控制　　C．过程支持　　D．质量控制

3.　某程序的程序图如下图所示，运用 McCabe 度量法对其进行度量，其环路复杂度是______。(2010 年上半年试题 36)





A．4　　B．5　　C．6　　D．8

4. 确定构建软件系统所需要的人数时，无须考虑______。(2009 年下半年试题 17)

A．系统的市场前景　　B．系统的规模

C．系统的技术复杂性　　D．项目计划

5. 风险预测从两个方面评估风险，即风险发生的可能性以及_____。(2009 年下半年试题 19)

A．风险产生的原因　　B．风险监控技术

C．风险能否消除　　D．风险发生所产生的后果

5.6.4 同步练习参考答案

(1) B　(2) D　(3) C　(4) A　(5) D

5.7 软件项目管理

5.7.1 考点辅导

5.7.1.1 软件项目管理涉及的范围

有效的软件项目管理集中在 4 个 P 上，即人员(Person)、产品(Product)、过程(Procedure)和项目(Project)。

5.7.1.2 软件项目估算

软件项目估算涉及人、技术、环境等多种因素，因此很难在项目完成前准确地估算出开发软件所需的成本、持续时间和工作量。因此，需要一些方法和技术来支持项目的估算，常用的估算方法有下列 3 种。

(1) 基于已经完成的类似项目进行估算。这是一种常用的也是有效的估算方法。

(2) 基于分解技术进行估算。分解技术包括问题分解和过程分解。问题分解是将一个复杂问题分解成若干个小问题，通过对小问题的估算得到复杂问题的估算。过程分解是指先根据软件开发过程中的活动(分析、设计、编码、测试等)进行估算，然后得到整个项目的估算值。

(3) 基于经验估算模型的估算。典型的经验估算模型有 IBM 估算模型、CoCoMo 模型和 Putnam 模型。

5.7.1.3 进度管理

软件开发项目的进度安排有两种方式：系统最终交付日期已经确定，软件开发部门必须在规定期限内完成；系统最终交付日期只确定了大致的年限，最后交付日期由软件开发部门确定。

进度安排的常用图形描述方法有甘特(Gantt)图和计划评审技术(PERT)图。

1)　Gantt 图

Gantt 图中横坐标表示时间，纵坐标表示任务，图中的水平线段则表示对一个任务的进度安排，线段的起点和终点对应在横坐标上的时间分别表示该任务的开始时间和结束时间，线段的长度表示完成该任务所需的时间。

Gantt 图能清晰地描述每个任务从何时开始，到何时结束以及各个任务之间的并行性；但是它不能清晰地反映出各任务之间的依赖关系，难以确定整个项目的关键所在，也不能反映计划中有潜力的部分。

2)　PERT 图

PERT 图是一个有向图，箭头表示任务，它可以表示完成该任务所需的时间；箭头指向节点表示流入节点的任务的结束，并开始流出节点的任务，这里把节点当成事件。只有当流入该节点的所有任务都结束时，节点所表示的事件才出现，流出节点的任务才可以开始。事件本身不消耗时间和资源，它仅表示某个时间点。一个事件有一个事件号和出现该事件的最早时刻和最迟时刻。每个任务还有一个松弛时间，表示在不影响整个工期的前提下，完成该任务有多少机动余地。

PERT 图不仅给出了每个任务的开始时间、结束时间和完成该任务所需的时间，还给出了任务之间的关系，即哪些任务完成后才能开始另外一些任务，以及如期完成整个工程的关键路径。松弛时间则反映了完成某些任务是可以推迟其开始时间或延长其所需的完成时间。但是 PERT 图不能反映任务之间的并行关系。

5.7.1.4　软件项目组织

开发组织采用什么形式组织，不仅要考虑软件项目的特点，还需要考虑参与人员的素质。在软件项目组织中，其组织原则有以下 3 条。

(1)　尽早落实责任。在软件项目开始组织时，要尽早指定专人负责，使他有权进行管理，并对任务的完成负全责。

(2)　减少交流接口。一个组织的生产率随着完成任务时存在的通信路径数目的增加而降低。要有合理的人员分工、好的组织结构、有效的通信，减少不必要的生产率的损失。

(3)　责权均衡。软件管理人员承担的责任不应比赋予他的权利还大。

5.7.1.5　软件配置管理

软件配置管理(Software Configure Management，SCM)用于整个软件工程过程。其目标是标识变更、控制变更、确保变更正确地实现、报告有关变更。SCM 是一组管理整个软件生存期各阶段中变更的活动。

1)　基线

基线是软件生存期中各开发阶段的一个特定点，它的作用是把开发各阶段工作的划分更加明确化，使本来连续的工作在这些点上断开，以便于检查与肯定阶段成果。因此，基线作为一个检查点，在开发过程中，当采用的基线发生错误时，可以知道所处的位置，返回到最近和最恰当的基线上。

2)　软件配置项

软件配置项(SCI)是软件工程中产生的信息项，它是配置管理的基本单位，对已经成为基线的 SCI，虽然可以修改，但必须按照一个特殊的、正式的过程进行评估，确认每一处

修改。

3) 版本控制

表达系统不同版本的一种表示方法如图 5-1 所示。

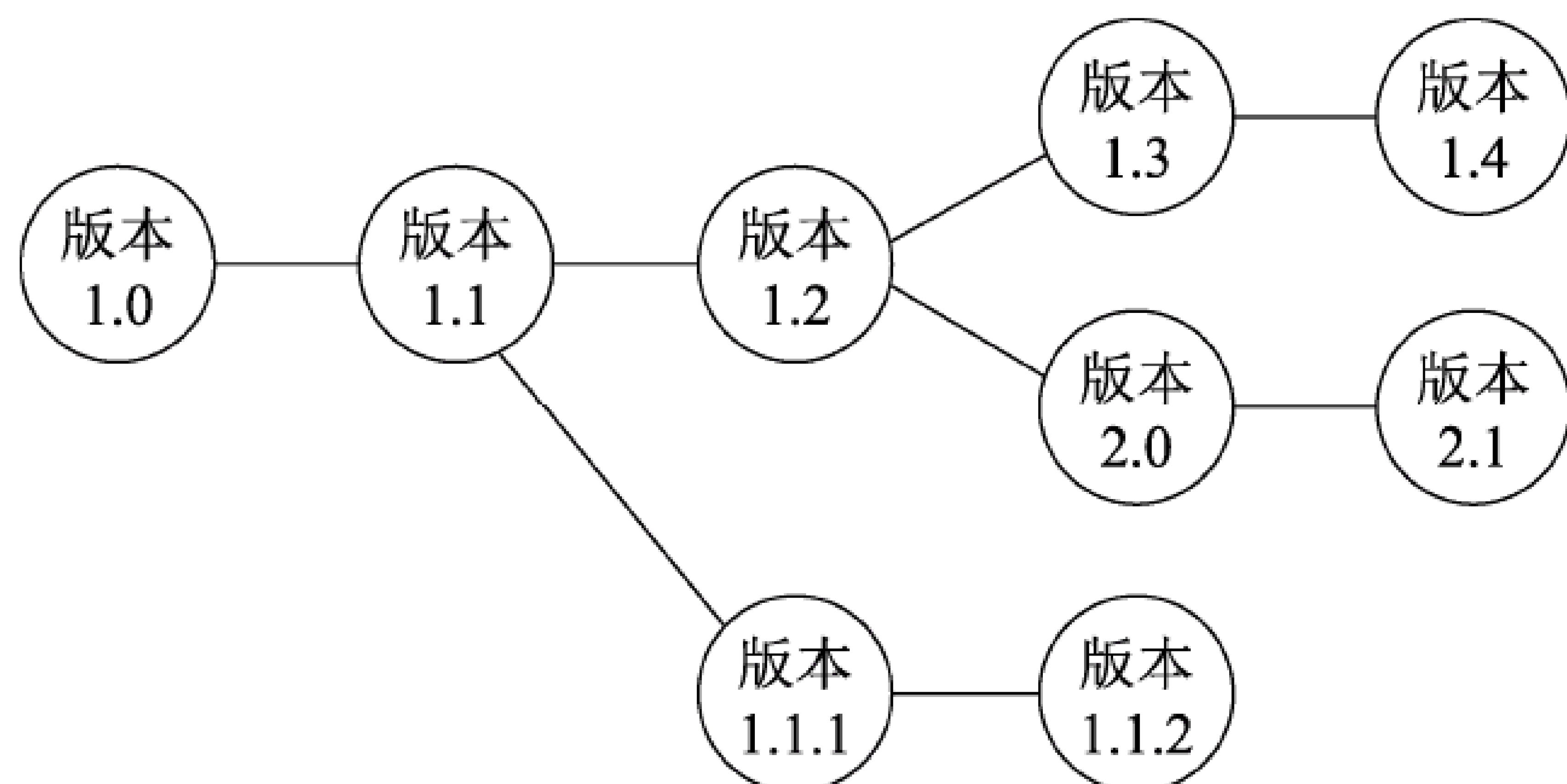


图 5-1　系统不同版本的一种表示方法

4) 变更控制

变更控制是一项最重要的软件配置任务。为有效地实现变更控制，须借助配置数据库和基线的概念。

配置数据库可以分为 3 类，即开发库、受控库和产品库。

5.7.1.6　风险分析

1) 风险识别

风险识别是试图系统化地确定对项目计划的威胁。风险识别的一个方法是建立风险条目检查表。该检查表可以用于识别风险，并使得人们集中来识别下列常见的、已知的及可预测的风险：产品规模、商业影响、客户特性、过程定义、开发环境等。

2) 风险预测

风险预测又称为风险估算，它从两个方面评估一个风险：风险发生的可能性或概率；以及如果风险发生所产生的后果。通常项目计划人员与管理人员、技术人员一起，进行 4 种风险预测活动：建立一个尺度或标准，以反映风险发生的可能性；描述风险的后果；估计风险对项目和产品的影响；标注风险预测的整体精确度，以免产生误解。

3) 风险评估

在进行风险评估时，建立了以下形式的三元组，即

$$(r_i, l_i, x_i)$$

式中：r_i 为风险；l_i 为风险发生的概率；x_i 为风险产生的影响。

一个对风险评估很有用的技术就是定义风险参照水准。对于大多数软件项目来说，成本、进度和性能就是 3 种典型的风险参照水准。

在风险评估过程中，需要执行下面步骤：①定义项目的风险参考水平值；②建立每一组与每一个参考水平值之间的关系；③预测一组临界点以定义项目终止区域；④预测什么样的风险组合会影响参考水平值。

4) 风险控制

一个有效的策略必须考虑 3 个问题：①风险避免；②风险监控；③风险管理及意外事

件计划。如果软件项目组对于风险采取主动的方法，则避免是最好的策略。

5.7.2　典型例题分析

例 1　定义风险参照水准是__(19)__活动常用的技术。(2012 年下半年试题 19)

(19) A．风险识别　　B．风险预测　　C．风险评估　　D．风险控制

解析：一个对风险评估很有用的技术就是定义风险参照水准。对于大多数软件项目来说，成本、进度和性能就是 3 种典型的风险参照水准。

答案：C

例 2　以下关于软件项目管理中人员管理的叙述，正确的是__(19)__。(2015 年下半年试题 19)

(19) A．项目组成员的工作风格也应该作为组织团队时要考虑的一个要素

B．鼓励团队的每个成员充分地参与开发过程的所有阶段

C．仅根据开发人员的能力来组织开发团队

D．若项目进度滞后于计划，则增加开发人员一定可以加快开发进度

解析：在软件项目中开发人员管理是核心的资源，其中人员的配置、调度安排贯穿整个软件项目过程中。人员安排的组织管理是否得当，对软件项目成功起到决定性的作用。在软件项目初始阶段，要根据工作量大小、所需的专业技能类型、团队成员能力水平、性格和开发经验，组建开发小组。整个项目被分解，项目中的成员根据所述的专业组的职能承担项目的相应任务。当项目进度滞后于计划时，下意识的反应往往是增加人力，这是不太可取的，因为在项目中新加入的程序员往往更难融入到项目中，所花费的时间代价会更大。

答案：A

例 3　若软件项目组对风险采用主动的控制方法，则__(19)__是最好的风险控制策略。(2012 年上半年试题 19)

(19) A. 风险避免　　B. 风险监控

C. 风险消除　　D. 风险管理及意外事件计划

解析：如果软件项目组对于风险采取主动的方法，则避免永远是最好的策略。这可以通过建立一个风险缓解计划来达到。例如，频繁的人员流动被标注为一个项目风险，基于以往的历史和管理经验，人员流动的概率为 70%，而影响被预测为对于项目成本及进度有严重的影响。为了缓解这个风险，项目管理者必须建立一个策略来降低人员流动。

答案：A

例 4　以下关于进度管理工具 Gantt 图的叙述中，不正确的是__(18)__。(2014 年上半年试题 18)

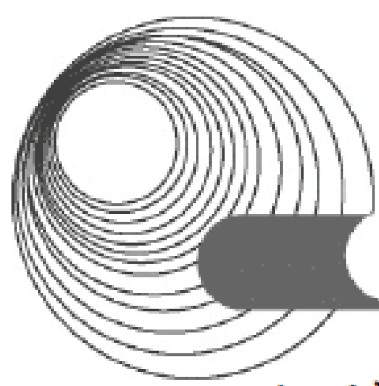
(18) A．能清晰地表达每个任务的开始时间、结束时间和持续时间

B．能清晰地表达任务之间的并行关系

C．不能清晰地确定任务之间的依赖关系

D．能清晰地确定影响进度的关键任务

解析：甘特图的思路简单，即以图示的方式通过活动列表和时间刻度形象地表示出任



何特定项目的活动顺序与持续时间。基本是一条线条图，横轴表示时间，纵轴表示活动(项目)，线条表示在整个期间上计划和实际的活动完成情况。它直观地表明任务计划在什么时候进行，以及实际进展与计划要求的对比。管理者由此可便利地弄清一项任务(项目)还剩下哪些工作要做，并可评估工作进度。

优点：能清晰地描述每个任务从何时开始，到何时结束以及各个任务之间的并行性。

缺点：不能清晰地反映出各任务之间的依赖关系，难以确定整个项目的关键所在，也不能反映计划中有潜力的部分。

答案：D

例 5 以下关于风险管理的叙述中，不正确的是 (19) 。(2011 年下半年试题 19)

(19) A. 仅根据风险产生的后果来对风险排优先级
B. 可以通过改变系统性能或功能需求来避免某些风险
C. 不可能去除所有风险，但可以通过采取行动来降低或减少风险
D. 在项目开发过程中，需要定期地评估和管理风险

解析：风险管理中包括了对风险的量度、评估和应变策略。理想的风险管理，是一连串排好优先次序的过程，使当中的可以引致最大损失及最可能发生的事情优先处理，而相对风险较低的事情则押后处理。现实情况中，优化的过程往往很难决定，因为风险和发生的可能性通常并不一致，所以要权衡两者的比重，以便作出最合适的决定。

答案：A

例 6 某个项目在开发时采用了不成熟的前沿技术，由此而带来的风险属于 (17) 风险。(2014 年下半年试题 17)

(17) A. 市场　B. 技术　C. 经济　D. 商业

解析：技术不成熟属于技术范围的风险。

答案：B

例 7 (17) 软件成本估算模型是一种静态单变量模型，用于对整个软件系统进行估算。(2014 年上半年试题 17)

(17) A. Putnam　B. 基本 COCOMO
C. 中级 COCOMO　D. 详细 COCOMO

解析：COCOMO 用 3 个不同层次的模型来反映不同程度的复杂性，它们分别如下。

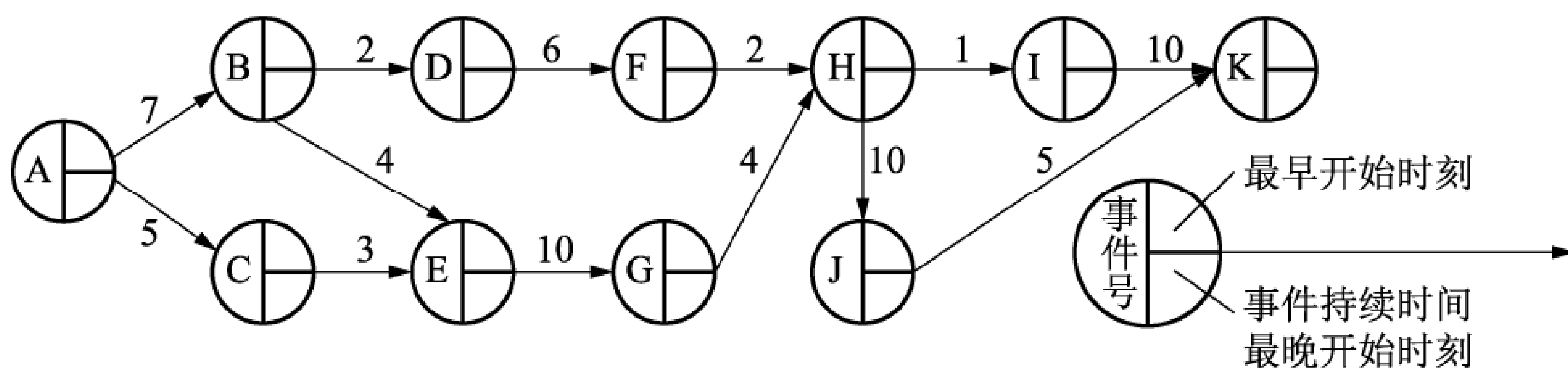
基本模型(Basic Model)：是一个静态单变量模型，它用一个以已估算出来的源代码行数(LOC)为自变量的函数来计算软件开发工作量。

中级模型(Intermediate Model)：在用 LOC 为自变量的函数计算软件开发工作量的基础上，再用涉及产品、硬件、人员、项目等方面属性的影响因素来调整工作量的估算。

详细模型(Detailed Model)：包括中级 COCOMO 型的所有特性，但用上述各种影响因素调整工作量估算时，还要考虑对软件工程过程中分析、设计等各步骤的影响。

答案：B

例 8 使用 PERT 图进行进度安排，不能清晰地描述 (16) ，但可以给出哪些任务完成后才能开始另一些任务。下图所示工程从 A～K 的关键路径是 (17) ，(图中省略了任务的开始和结束时刻)。(2010 年下半年试题 16、17)



(16) A．每个任务从何时开始　　B．每个任务到何时结束

C．各任务之间的并行情况　　D．各任务之间的依赖关系

(17) A．ABEGHIK　　B．ABEGHJK　　C．ACEGHIK　　D．ACEGHJK

解析：PERT 图给出了每个任务的开始时间、结束时间和完成该任务所需要的时间，同时还给出了任务之间的依赖关系，即哪些任务完成后才能执行另外一些任务。PERT 图的不足是不能反映任务之间的并行关系。

关键路径是松弛时间为 0 的任务完成过程所经历的路径。本题的图中没有给出松弛时间，因此关键路径是耗时最长的路径，即 A→B→E→G→H→J→K。

答案：(16) C　　(17) B

例 9　在采用面向对象技术构建软件系统时，很多敏捷方法都建议的一种重要的设计活动是__(15)__，它是一种重新组织的技术，可以简化构件的设计而无须改变其功能或行为。(2009 年上半年试题 15)

(15) A．精化　　B．设计类　　C．重构　　D．抽象

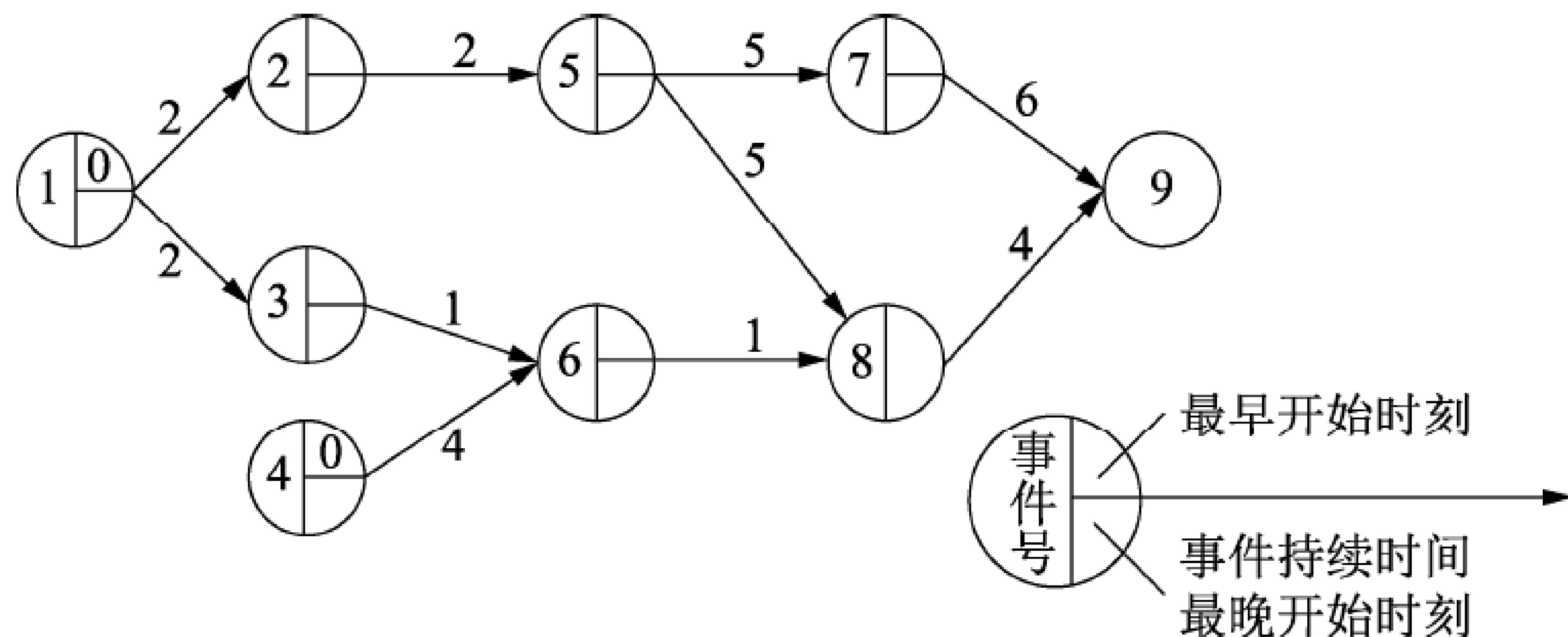
解析：本题考查软件过程管理中的敏捷方法。

在敏捷方法中重构是指重新审视需求和设计，重新明确地描述它们以符合新的和现有的需求。而抽象是为了简化构件的设计且无须改变其功能。

答案：D

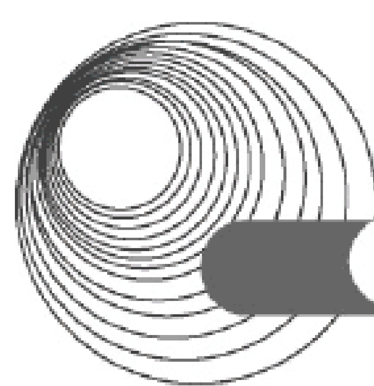
5.7.3　同步练习

1.　进度安排的常用图形描述方法有 Gantt 图和 PERT 图。Gantt 图不能清晰地描述__(1)__；PERT 图可以给出哪些任务完成后才能开始另一些任务。下图所示的 PERT 图中，事件 6 的最晚开始时刻是__(2)__。(2010 年上半年试题 18、19)



(1)　A．每个任务从何时开始　　B．每个任务到何时结束

C．每个任务的进展情况　　D．各任务之间的依赖关系



(2) A. 0　　B. 3　　C. 10　　D. 11

2. 能力成熟度集成模型 CMMI 是 CMM 模型的最新版本，它有连续式和阶段式两种表示方式。基于连续式表示的 CMMI 共有 6 个(0～5)能力等级，每个能力等级对应到一个一般目标以及一组一般执行方法和特定方法，其中能力等级__(3)__主要关注过程的组织标准化和部署。(2010 年上半年试题 31)

A. 1　　B. 2　　C. 3　　D. 4

5.7.4 同步练习参考答案

答案：

(1) D　(2) C　(3)C

5.8 软 件 质 量

5.8.1 考点辅导

软件质量是指反映软件系统或软件产品满足规定或隐含要求的能力的特征和特性全体。软件质量保证是为保证软件系统或软件产品充分满足用户要求的质量而进行的有计划、有组织的活动，其目的是生产该质量的软件。

5.8.1.1 软件质量特性

ISO/IEC 9126 软件质量模型由 3 个层次组成：第一层是质量特性；第二层是质量子特性；第三层是度量指标。该模型的质量特性和质量子特性的含义如下。

(1) 功能性。与一组功能及其指定的性质的存在有关的一组属性。功能是指能满足规定或隐含需求的那些功能。

- 适合性：与规定任务能否提供一组功能以及这组功能能否适合有关的软件属性。
- 准确性：与能否得到正确的或相符的结果或效果有关的软件属性。
- 互用性：与同其他指定系统进行交互操作的能力有关的软件属性。
- 依从性：使软件服从有关的标准、约定、法规及类似规定的软件属性。
- 安全性：与避免对程序及数据的非授权故意或意外访问的能力有关的软件属性。

(2) 可靠性。与在规定的一段时间内和规定的条件下，软件维持其性能水平有关的能力。

- 成熟性：与由软件故障引起失效的频度有关的软件属性。
- 容错性：与在软件错误或违反指定接口情况下，维持指定的性能水平的能力有关的软件属性。
- 易恢复性：与在故障发生后重新建立其性能水平并恢复直接受影响数据的能力，以及为达此目的所需的时间有关的软件属性。

(3) 易使用性。与为使用软件所需的努力和由一组规定的或隐含的用户对如此使用所做的评价有关的一组属性。

- 易理解性：与用户为理解逻辑概念及其应用范围所花的努力有关的软件属性。
- 易学性：与用户为学习其应用(如操作控制、输入、输出)所需努力有关的软件属性。
- 易操作性：与用户为进行操作或操作控制所需努力有关的软件属性。

(4) 效率。与在规定条件下，软件的性能水平与所用资源量之间的关系有关的一组属性。

- 时间特性：与响应和处理时间以及软件执行其功能时的吞吐量有关的软件属性。
- 资源特性：与软件执行其功能时所使用的资源量以及使用资源的持续时间有关的软件属性。

(5) 可维护性。与进行规定的修改所需努力有关的一组属性。

- 易分析性：与为诊断缺陷或失效原因，或为判定待修改的部分所需努力有关的软件属性。
- 易改变性：与进行修改、调试或适应环境变化所需努力有关的软件属性。
- 稳定性：与修改造成未预料后果的风险有关的软件属性。
- 易测试性：与确认修改软件所需努力有关的软件属性。

(6) 可移植性。与软件从一种环境转移到另一种环境的能力有关的一组属性。

- 适应性：与软件无须采用特别为该软件准备的处理手段就能适应规定的环境有关的软件属性。
- 易安装性：与在指定环境下安装软件所需努力有关的软件属性。
- 一致性：使软件服从与可移植性有关的标准或约定的软件属性。
- 易替换性：与软件在该软件环境中用来替代指定的其他软件的可能和努力有关的软件属性。

5.8.1.2　软件质量保证

软件质量保证包括与以下 7 个主要活动相关的各种任务。

(1) 应用技术方法。软件质量保证首先从一组技术方法和工具开始，这些方法和工具帮助分析人员形成高质量的规格说明和高质量的设计。

(2) 进行正式的技术评审。这是一种由技术人员实施的程式化会议，其唯一的目的是揭露质量问题。

(3) 测试软件。软件测试组合了多种测试策略，这些测试策略带有一系列有助于有效地检测错误的测试用例及设计方法。

(4) 标准的实施。多数情况下，标准由客户或某些章程确定。与标准是否一致的评估可以被软件开发者作为正式技术评审的一部分来进行。

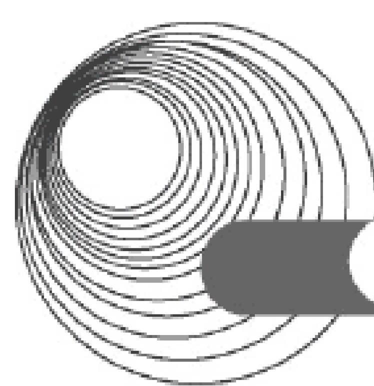
(5) 控制变更。变更控制过程通过对变更的正式申请、评价变更的特性和控制变更的影响等直接提高软件的质量。变更控制应用于软件开发期间和较后的软件维护阶段。

(6) 计量。其包括某些技术上的和面向管理的计量。

(7) 记录保存和报告。为软件质量保证提供收集和传播软件质量保证信息的过程。评审、监察、变更控制、测试和其他软件质量保证活动的结果必须变成项目历史记录的一部分，并且应当把它传播给需要知道这些结果的开发人员。

5.8.1.3　软件评审

通常把“质量”理解为“用户满意程度”。为了使得用户满意，有以下两个必要条件。



(1) 设计的规格说明书符合用户的要求，这称为设计质量。

(2) 程序按照设计规格说明所规定的情况正确执行，这称为程序质量。

设计质量评审的对象是在需求分析阶段产生的软件需求规格说明、数据需求规格说明，在软件概要设计阶段产生的软件概要设计说明书等。

程序质量评审通常是从开发者的角度进行评审，与开发技术直接相关。它是着眼于软件本身的结构、与运行环境的接口以及变更带来的影响而进行的评审活动。

5.8.1.4 软件容错技术

1) 容错软件定义

归纳容错软件的定义，有以下 4 种。

(1) 规定功能的软件，在一定程度上对自身错误的作用(软件错误)具有屏蔽能力，则称此软件为具有容错功能的软件，即容错软件。

(2) 规定功能的软件，在一定程度上能从错误状态自动恢复到正常状态，则称之为容错软件。

(3) 规定功能的软件，在发生错误时，仍然能在一定程度上完成预期的功能，则把该软件称为容错软件。

(4) 规定功能的软件，在一定程度上具有容错能力，则称之为容错软件。

2) 容错的一般方法

实现容错的主要手段是冗余。冗余是指对于实现系统规定功能是多余的那部分资源，包括硬件、软件、信息和时间。由于加入了这些资源，有可能使系统的可靠性得到较大的提高。通常冗余技术分为 4 类，即结构冗余、信息冗余、时间冗余和冗余附加技术。

5.8.2 典型例题分析

例 1 冗余技术通常分为 4 类，其中__(29)__按照工作方法可以分为静态、动态和混合冗余。(2010 年下半年试题 29)

(29) A. 时间冗余　　B. 信息冗余　　C. 结构冗余　　D. 冗余附加技术

解析：实现容错的主要手段是冗余。通常冗余技术分为 4 类，即结构冗余、信息冗余、时间冗余和冗余附加技术。

结构冗余是通常采用的冗余技术，按其工作方法可以分为静态、动态和混合冗余。

信息冗余通常采用奇偶码、循环码等冗余码制式以发现甚至纠正信息在运算或传输中出现的错误。

时间冗余是指以重复执行指令或程序来消除瞬时错误带来的影响。

冗余附加技术是指为实现上述冗余技术所需要的资源和技术，包括程序、指令、数据、存放和调动它们的空间和通道等。

答案：C

例 2 将每个用户的数据和其他用户的数据隔离开，是考虑了软件的__(31)__质量特性。(2011 年下半年试题 31)

(31) A. 功能性　　B. 可靠性　　C. 可维护性　　D. 易使用性

解析：“将每个用户的数据和其他用户的数据隔离开”是从安全性方面提出的功能要求，这主要是考虑了软件的功能性质量特性。

答案：A

例 3　正式技术评审的目标是__(33)__。(2015 年上半年试题 33)

(33) A. 允许高级技术人员修改错误

B. 评价程序员的工作效率

C. 发现软件中的错误

D. 记录程序员的错误情况并与绩效挂钩

解析：技术评审(Technical Review，TR)的目的是尽早地发现工作成果中的缺陷，并帮助开发人员及时消除缺陷，从而有效地提高产品的质量。技术评审包括：制定技术评审计划、正式技术评审、非正式技术评审。

答案：C

5.8.3　同步练习

软件维护成本在软件成本中占较大比重。为降低维护的难度，可采取的措施有__(1)__。

A. 设计并实现没有错误的软件

B. 限制可修改的范围

C. 增加维护人员数量

D. 在开发过程中就采取有利于维护的措施，并加强维护管理

5.8.4　同步练习参考答案

D

5.9　软 件 度 量

5.9.1　考点辅导

5.9.1.1　软件度量分类

软件度量有两种分类方法，第一种分类是将软件度量分为面向规模的度量、面向功能的度量和面向人的度量；第二种分类是将软件度量分为生产率度量、质量度量和技术度量。

软件生产率度量主要关注于软件工程活动的制品。软件质量度量可指明软件满足明确的和隐含的用户需求的程度。技术度量主要集中在软件产品的某些特征(如逻辑复杂性、模块化程度等)上，而不是软件开发的全过程。

面向规模的度量用于收集与软件规模相关的软件工程输出信息和质量信息，面向功能的度量则集中在程序的“功能性”和“实用性”。面向人的度量收集有关人们开发软件所

用方式的信息和人员理解有关工具的方法和效率的信息。还有基于问题、基于过程、基于用例等成本估算方法。

5.9.1.2 软件复杂性度量

软件复杂性是指理解和处理软件的难易程度。软件复杂性度量的参数很多，主要有以下几个。

规模。规模即总共的指令数，或源程序行数。

难度。通常由程序中出现的操作数的数目所决定的量来表示。

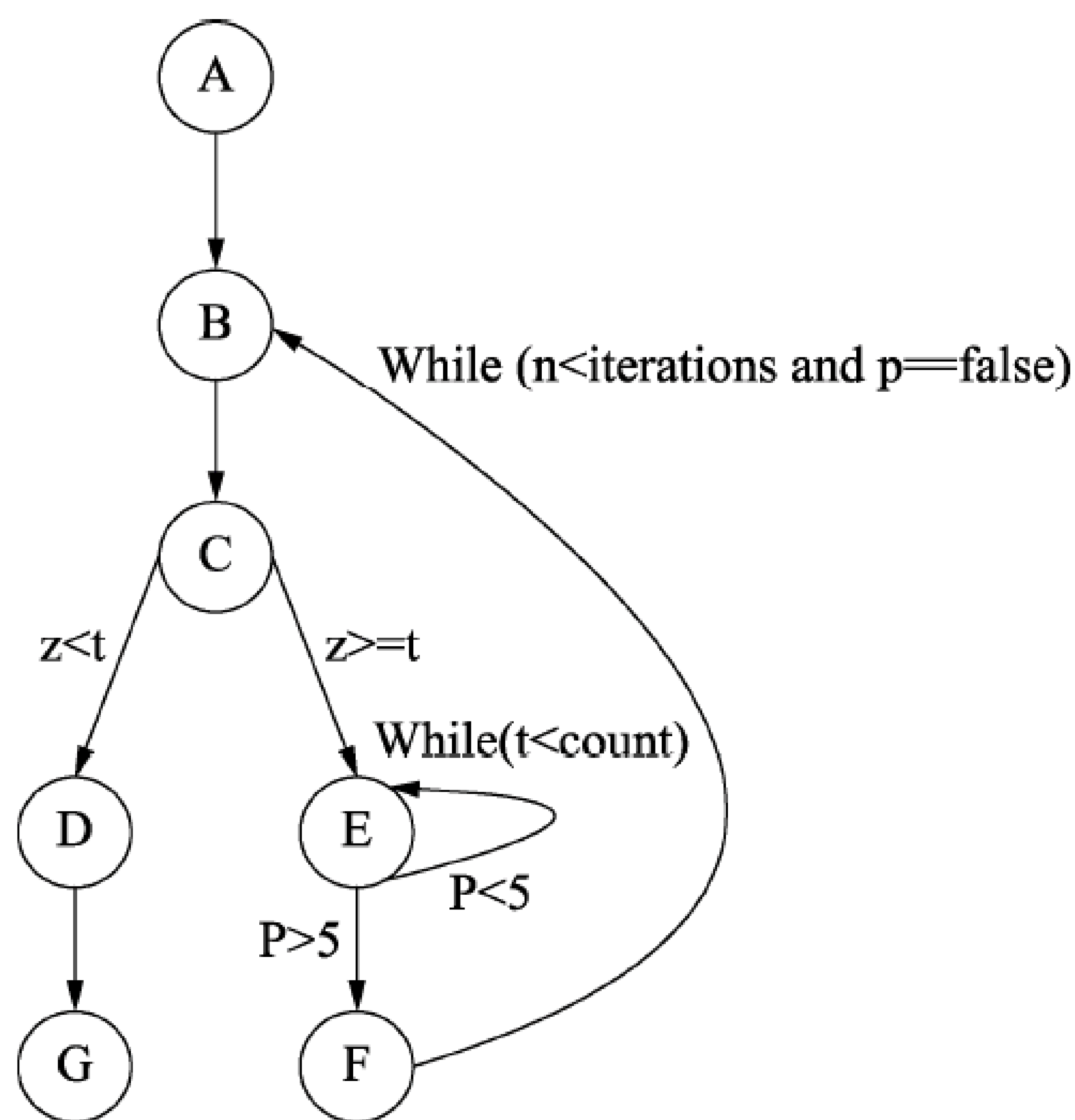
结构。通常用与程序结构有关的度量来表示。

智能度。智能度即算法的难易程度。

软件复杂性包括程序复杂性和文档复杂性，软件复杂性主要体现在程序的复杂性中。

5.9.2 典型例题分析

例1 McCabe 度量法是通过定义环路复杂度，建立程序复杂性的度量，它基于一个程序模块的程序图中环路的个数。计算有向图 G 的环路复杂性的公式为 $V(G)=m-n+2$，其中 $V(G)$是有向图 G 中的环路个数，m 是 G 中的有向弧数，n 是 G 中的节点数。下图所示程序图的程序复杂度是<u>(32)</u>。(2009年上半年试题32)



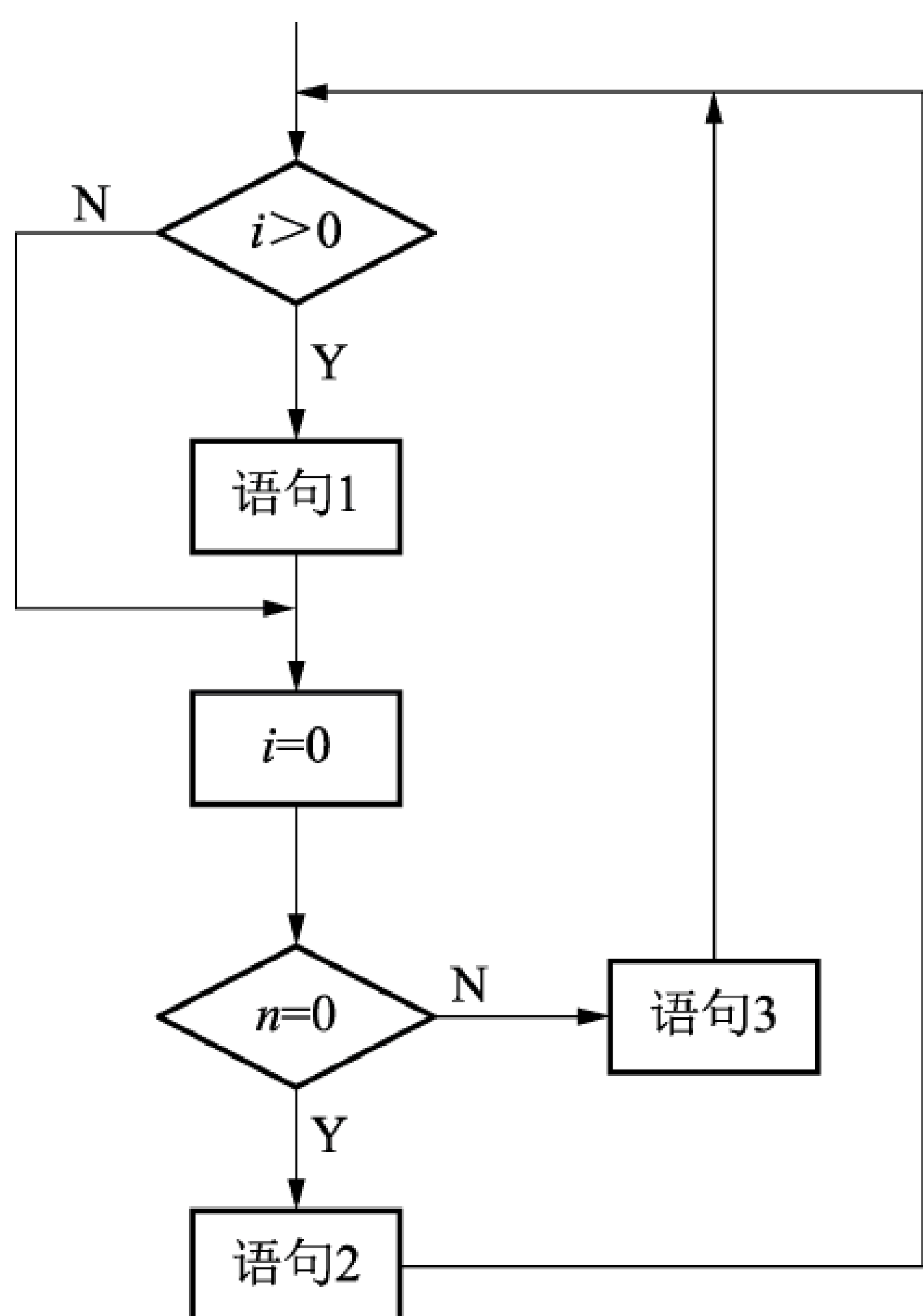
(32) A．2　　B．3　　C．4　　D．5

解析： 方法一，图中有分支的节点个数为2个，从而环路复杂度为2+1=3。

方法二，依据题中给的公式，图中有向弧数 m 为 8 个，图中节点数 n 为 7 个。故 $V(G)=m-n+2=3$。

答案： B

例2 采用 McCabe 度量法计算下图环路复杂性为<u>(31)</u>。(2012年下半年试题31)

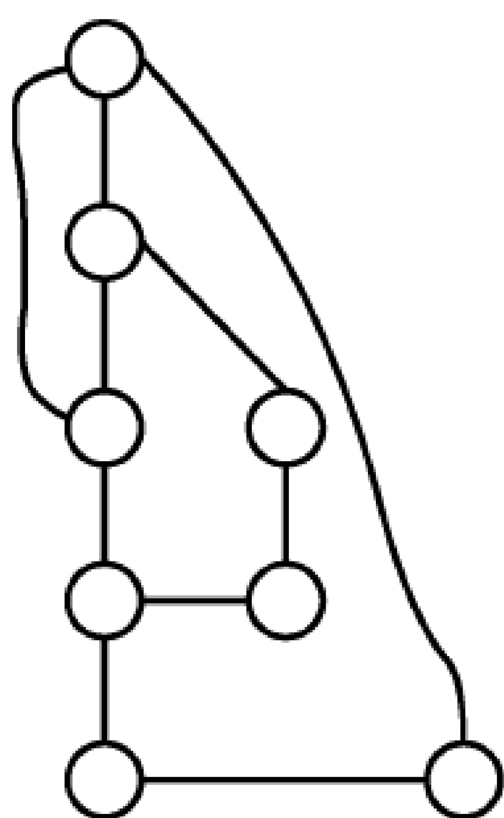


(31) A．2　　　　B．3　　　　C．4　　　　D．5

解析：McCabe 度量法是一种基于程序控制流的复杂性度量方法(度量也称计量)，该方法认为程序的复杂性很大程度上取决于控制的复杂性。首先根据程序画出程序图，然后基于图论用图的环路数来度量程序复杂性，即 $V(G)=m-n+2p$，其中 m、n 和 p 分别表示图 G 中弧的个数、顶点的个数和强连通分量数。由上图可知，$m=8$，$n=6$，$p=1$，根据上述公式可得，其环路复杂性为 8-6+2=4。

答案：C

例 3　采用 McCabe 度量法计算下列程序图的环路复杂性为<u>(35)</u>。(2015 年上半年试题 35)

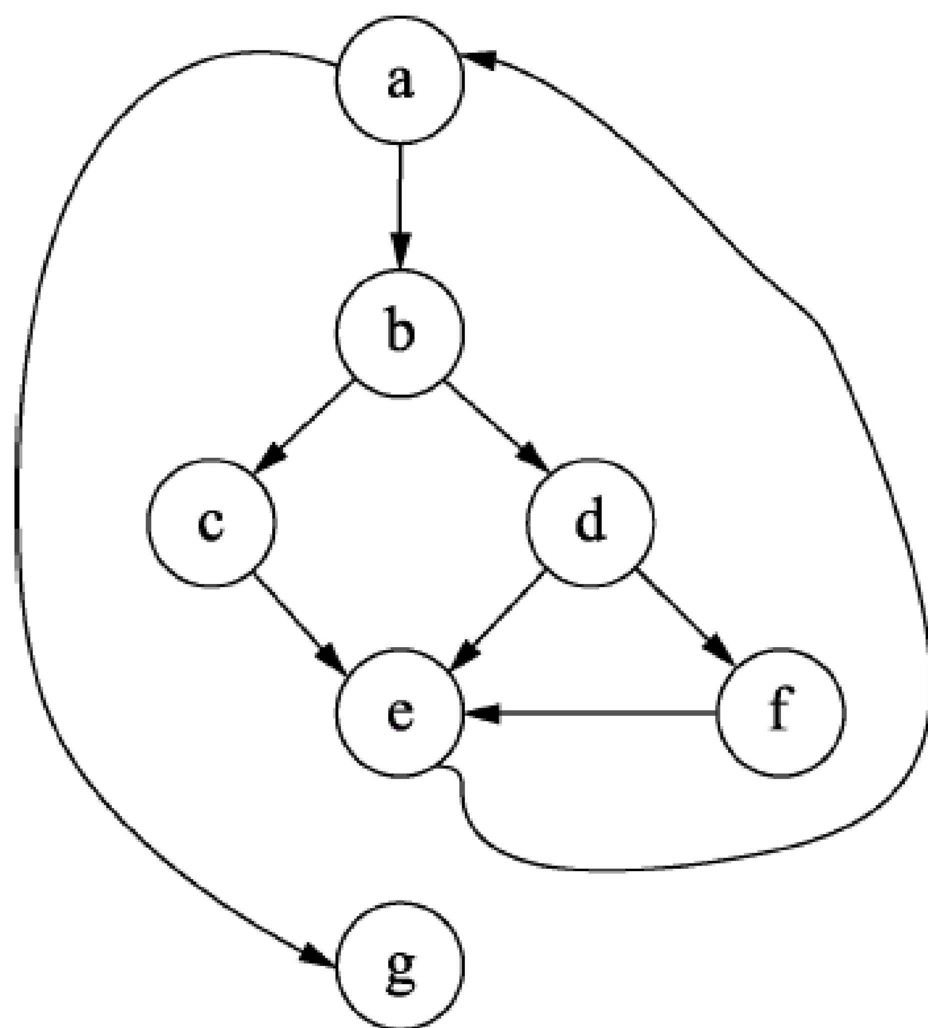


(35) A. 2　　　　B. 3　　　　C. 4　　　　D. 5

解析：程序图的节点数为 8，弧数为 10，因此环路复杂度为 10-8+2=4。

答案：C

例 4　根据 McCabe 度量法，下图所示程序图的复杂性度量值为<u>(32)</u>。(2010 年下半年试题 32)



(32) A．4　　　　B．5　　　　C．6　　　　D．7

解析：对于强连通的有向图，复杂性度量值为 $m-n+2p$，其中 m 是图中弧的个数，n 是图中的节点数，p 是图中强连通分量的个数。

题图中弧的个数为 9，节点的个数为 7，强连通分量的个数为 1，因此，可以计算题图的复杂性度量值为 4。

答案：A

5.9.3　同步练习

软件复杂性度量的参数不包括__(1)__。

(1) A．软件的规模　　　　B．开发小组的规模
　　C．软件的难度　　　　D．软件的结构

5.9.4　同步练习参考答案

(1)　B

5.10　软件工具与软件开发环境

5.10.1　考点辅导

5.10.1.1　软件开发工具

软件工具可以分为以下几类。

(1) 软件开发工具。其包括需求分析工具、设计工具和编码与排错工具。

(2) 软件维护工具。其包括版本控制工具、文档分析工具、开发信息库工具、逆向工程工具和再工程工具。

(3) 软件管理和支持工具。其包括项目管理工具、配置管理工具和软件评价工具。

5.10.1.2　软件开发环境

软件开发环境是支持软件产品开发的软件系统。它由软件工具集和环境集成机制构成，前者用来支持软件开发的相关过程、活动和任务等；后者为工具集成和软件开发、维护和管理提供统一的支持，它通常包括数据集成、控制集成和界面集成。

软件开发环境的特征是：环境的服务是集成的；环境应支持小组工作方式，并为其提供配置管理；环境的服务可用于支持各种软件开发活动。

集成型开发环境是一种把支持多种软件开发方法和开发模型的软件工具集成在一起的软件开发环境。这种环境具有开放性和可剪裁性。开放性为环境外的工具集成到环境中来提供了方便，可剪裁性可根据不同的应用和不同的用户需求进行剪裁，以形成特定的开发环境。

5.10.2　典型例题分析

例　对于一个大型软件来说，不加控制的变更很快就会引起混乱。为有效地实现变更控制，需借助配置数据库和基线的概念。__(29)__ 不属于配置数据库。(2010 年上半年试题 29)

(29) A．开发库　　B．受控库　　C．信息库　　D．产品库

解析：软件变更控制是变更管理的重要内容，要有效地进行变更控制，需要借助配置数据库和基线的概念。配置数据库一般包括开发库、受控库和产品库。

答案：C

5.10.3　同步练习

在系统转换的过程中，旧系统和新系统并行工作一段时间，再由新系统代替旧系统的策略称为 __(1)__；在新系统全部正式运行前，一部分一部分地代替旧系统的策略称为 __(2)__。

(1)～(2) A．直接转换　　B．位置转换
C．分段转换　　D．并行转换

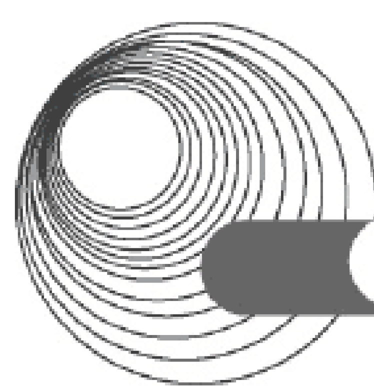
5.10.4　同步练习参考答案

(1)D　　(2)C

5.11　本 章 小 结

本章知识点在 2013 年的新大纲中改动不大，只是有一些描述方面的调整。

本章主要要求考生掌握软件工程和软件项目管理的基本知识和概念，包括软件工程知识、系统分析基础知识、系统设计知识、系统实施知识、系统运行和维护知识等内容。



软件工程知识是历年软件设计师考试重点考查的内容之一，从 2009—2013 年这四年八次考试看，这个章节在上午试题中一般考 11 道小题左右，因此要系统掌握软件工程的全部内容，树立软件工程的观念，并将其作为指导软件设计和分析活动的理论基础。

5.12 达标训练题及参考答案

5.12.1 达标训练题

1. 软件能力成熟度模型(Capability Maturity Model，CMM)描述和分析了软件过程能力的发展与改进的程度，确立了一个软件过程成熟程度的分级标准。在初始级，软件过程定义几乎处于无章可循的状态，软件产品的成功往往依赖于个人的努力和机遇。

在__(1)__，已建立了基本的项目管理过程，可对成本、进度和功能特性进行跟踪。

在__(2)__，用于软件管理与工程两方面的软件过程均已文档化、标准化，并形成了整个软件组织的标准软件过程。在已管理级，对软件过程和产品质量有详细的度量标准。

在(3)，通过对来自过程、新概念和新技术等方面的各种有用信息的定量分析，能够不断地、持续地对过程进行改进。

(1) A．可重复级　B．管理级　C．功能级　D．成本级

(2) A．标准级　B．已定义级　C．可重复级　D．优化级

(3) A．分析级　B．过程级　C．优化级　D．管理级

2. 概要设计是软件系统结构的总体设计，以下选项中不属于概要设计的是________。

A．把软件划分成模块　B．确定模块之间的调用关系

C．确定各个模块的功能　D．设计每个模块的伪代码

3. 软件设计中划分模块的一个准则是__(1)__。两个模块之间的耦合方式中，__(2)__耦合的耦合度最高，__(3)__耦合的耦合度最低。一个模块内部的内聚种类中，__(4)__内聚的内聚度最高，__(5)__内聚的内聚度最低。

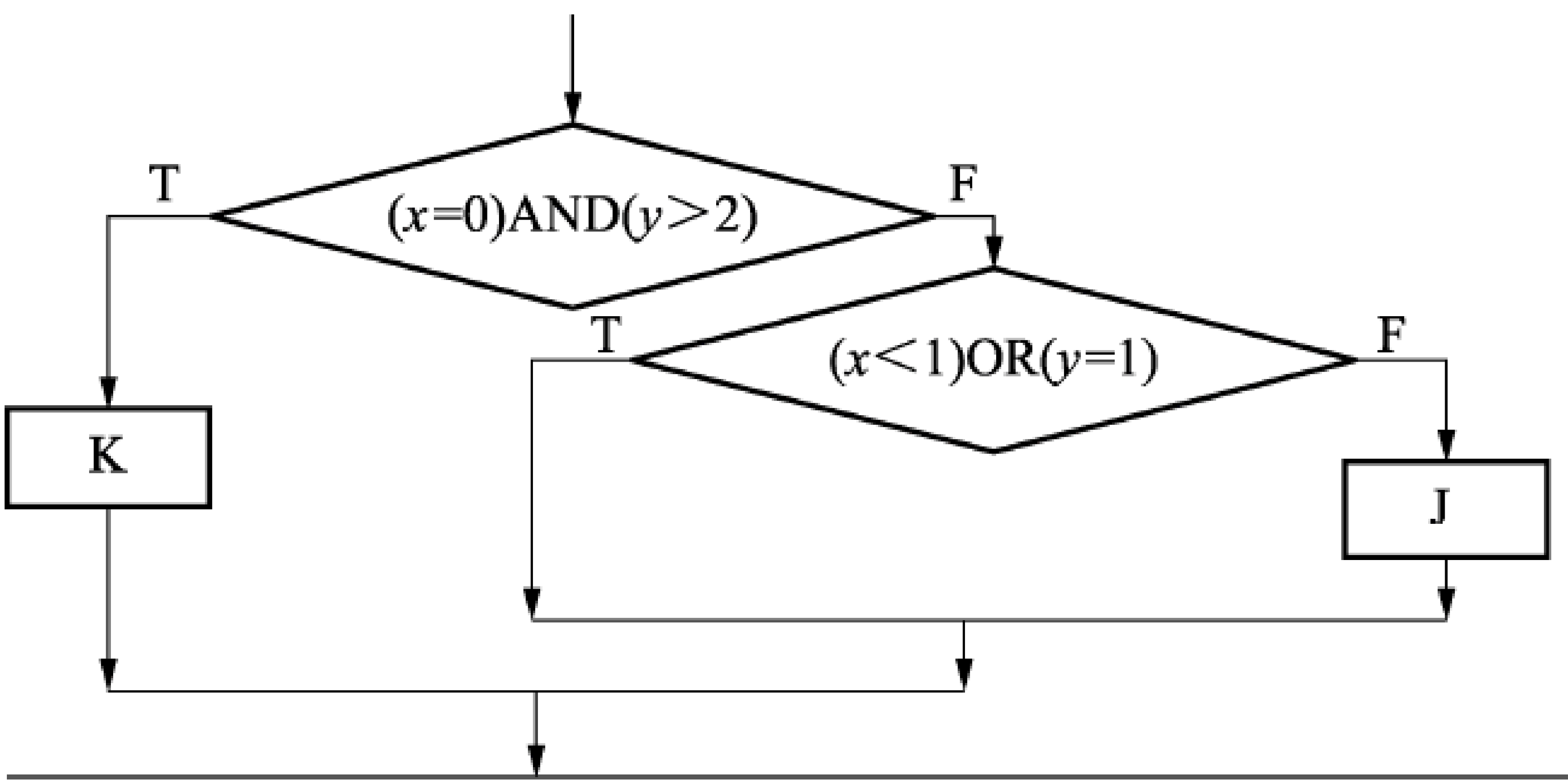
(1) A．低内聚低耦合　B．低内聚高耦合

C．高内聚低耦合　D．高内聚高耦合

(2)～(3) A．数据　B．非直接　C．控制　D．内容

(4)～(5) A．偶然　B．逻辑　C．功能　D．过程

4．在结构测试用例设计中，有语句覆盖、条件覆盖、判定覆盖(即分支覆盖)、路径覆盖等，其中__(1)__是最强的覆盖准则。为了对下图所示的程序段进行覆盖测试，必须适当地选取测试数据组。若 x、y 是两个变量，可供选择的测试数据组共有Ⅰ、Ⅱ、Ⅲ、Ⅳ四组(如下表中给出)，则实现判定覆盖至少应采用的测试数据组是__(2)__；实现条件覆盖至少应采用的测试数据组是__(3)__；实现路径覆盖至少应采用的测试数据组是__(4)__或__(5)__。



	x	y
测试数组 I	0	3
测试数组 II	1	2
测试数组III	−1	2
测试数组IV	3	1

(1) A．语句覆盖　　B．条件覆盖　　C．判定覆盖　　D．路径覆盖

(2)～(5) A．Ⅰ和Ⅱ组　　B．Ⅱ和Ⅲ组　　C．Ⅲ和Ⅳ组

D．Ⅰ和Ⅳ组　　E．Ⅰ、Ⅱ和Ⅲ组　　F．Ⅱ、Ⅲ和Ⅳ组

G．Ⅰ、Ⅲ和Ⅳ组　　H．Ⅰ、Ⅱ和Ⅳ组

5.12.2　参考答案

1．(1) A　(2) B　(3) C

2．D

3．(1) C　(2) D　(3) B　(4) C　(5) A

4．(1) D　(2) A　(3) D　(4) E　(5) H

第6章　结构化开发方法

大纲要求：

- 结构化方法由结构化分析、结构化设计、结构化程序设计构成，它是一种面向数据流的开发方法。结构化分析是根据分解与抽象的原则，按照系统中数据处理的流程，用数据流图来建立系统的功能模型，从而完成需求分析工作。结构化设计是根据模块独立性准则、软件结构优化准则将数据流图转换为软件的体系结构，用软件结构图来建立系统的物理模型，实现系统的概要设计。结构化程序设计使用3种基本控制结构构造程序，任何程序都可以由顺序、选择和重复3种基本控制结构构造。
- 结构化方法总的指导思想是自顶向下、逐层分解，它的基本原则是功能的分解与抽象。它是软件工程中最早出现的开发方法，特别适合于数据处理领域的问题，但是不适合解决大规模的、特别复杂的项目，且难以适应需求的变化。

6.1　系统分析与设计概述

6.1.1　考点辅导

6.1.1.1　系统分析概述

系统分析是一种问题求解技术，它将一个系统分解成各个组成部分，目的是研究各个部分如何工作、交互，以实现其系统目标。系统分析的目的是为项目团队提供对触发项目的问题和需求的更全面的理解，因此强调业务问题方面，而非技术或实现方面。系统分析阶段要求和系统用户一起工作，以便清楚地定义新系统的业务需求和预期。

1. 系统分析的目的和任务

系统分析的主要任务是对现行系统进一步详细调查，将调查中所得到的文档资料集中，对组织内部整体管理状况和信息处理过程进行分析，为系统开发提供所需的资料，并提交系统方案说明书。系统分析侧重于从业务全过程的角度进行分析，主要内容有业务和数据的流程是否通畅、是否合理；数据、业务过程和组织管理之间的关系；原系统管理模式改革和新系统管理方法的实现是否具有可行性等。

确定的分析结果包括开发者对于现有组织管理状况的了解，用户对信息系统功能的需求，数据和业务流程，管理功能和管理数据指标体系以及新系统拟改动和新增的管理模型等。

最后，提出信息系统的各种设想和方案，并对所有的设想和方案进行分析、研究、比较、判断和选择，获得一个最优的新系统的逻辑模型，并在用户理解计算机系统的工

作流程和处理方式的情况下，将它明确地表达成书面资料——系统分析报告，即系统方案说明书。

2. 系统分析的步骤

企业信息系统是一个具有业务复杂性和技术复杂性的大系统，为的是目标系统既能实现当前系统的基本职能，又能改进和提高。系统开发人员首先必须理解并描述出已经实际存在的当前系统，然后进行改进，从而创造出基于当前系统又高于当前系统的目标系统，即新系统。

系统分析过程一般按如下步骤进行。

(1) 对当前系统进行详细调查，收集数据。

(2) 建立当前系统的逻辑模型。

(3) 对现状进行分析，提出改进意见和新系统应达到的目标。

(4) 建立新系统的逻辑模型。

(5) 编写系统方案说明书。

6.1.1.2　系统设计的基本原理

1. 抽象

抽象是一种设计技术，重点说明一个实体的本质方面，而忽略或者掩盖不太重要或非本质的方面。抽象是一种重要的工具，用来将复杂的现象简化到可以分析、实验或者可以理解的程度。软件工程中从软件定义到软件开发要经历多个阶段，在这个过程中每前进一步都可看作是对软件解法的抽象层次的一次细化。抽象的最底层就是实现该软件的源程序代码。在进行模块化设计时也可以有多个抽象层次，最高抽象层次的模块用概括的方式叙述问题的解法，较低抽象层次的模块是较高抽象层次模块对问题解法描述的细化。

2. 模块化

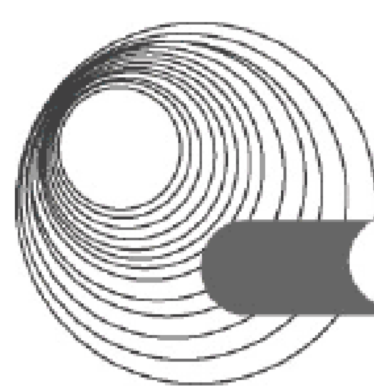
模块在程序中是数据说明、可执行语句等程序对象的集合，或者是单独命名和编址的元素，例如高级语言中的过程、函数和子程序等。在软件的体系结构中，模块是可组合、分解和更换的单元。

模块化是指将一个待开发的软件分解成若干个小的简单部分——模块，每个模块可独立地开发、测试，最后组装成完整的程序。这是一种复杂问题“分而治之”的原则。模块化的目的是使程序的结构清晰，容易阅读、理解、测试和修改。

3. 信息隐蔽

信息隐蔽是开发整体程序结构时使用的法则，即将每个程序的成分隐蔽或封装在一个单一的设计模块中，在定义每一个模块时尽可能少地显露其内部的处理。在设计时首先列出一些可能发生变化的因素，在划分模块时将一个可能发生变化的因素隐蔽在某个模块的内部，使其他模块与这个因素无关。当这个因素发生变化时，只需修改含有这个因素的模块，而与其他模块无关。

信息隐蔽原则对提高软件的可修改性、可测试性和可移植性都有重要的作用。



4. 模块独立

模块独立是指每个模块完成一个相对独立的特定子功能，并且与其他模块之间的联系简单。衡量模块独立程度的标准有两个：耦合性和内聚性。

1) 耦合

耦合是模块之间的相对独立性(互相连接的紧密程度)的度量。耦合取决于各个模块之间接口的复杂程度、调用模块的方式以及通过接口的信息类型等。一般模块之间可能的耦合方式有 7 种类型：

无直接耦合。指两个模块之间没有直接的关系，它们分别从属于不同模块的控制与调用，它们之间不传递任何信息。因此，模块间耦合性最弱，模块独立性最高。

数据耦合。指两个模块之间有调用关系，传递的是简单的数据值，相当于高级语言中的值传递。

标记耦合。指两个模块之间传递的是数据结构。

控制耦合。指一个模块调用另一个模块时，传递的是控制变量，被调用模块通过该控制变量的值有选择地执行模块内的某一功能。因此，被调用模块应具有多个功能，哪个功能起作用受调用模块控制。

外部耦合。模块间通过软件之外的环境联结(如 I/O 将模块耦合到特定的设备、格式、通信协议上)时称为外部耦合。

公共耦合。指通过一个公共数据环境相互作用的那些模块间的耦合。

内容耦合。当一个模块直接使用另一个模块的内部数据，或通过非正常入口转入另一个模块内部时，这种模块之间的耦合称为内容耦合。

2) 内聚

内聚是对一个模块内部各个元素彼此结合的紧密程度的度量。一个内聚程度高的模块(在理想情况下)应当只做一件事。一般模块的内聚性分为 7 种类型：

偶然内聚(巧合内聚)。指一个模块内的各处理元素之间没有任何联系。

逻辑内聚。指模块内执行若干个逻辑上相似的功能，通过参数确定该模块完成哪一个功能。

时间内聚。把需要同时执行的动作组合在一起形成的模块称为时间内聚模块。

过程内聚。指一个模块完成多个任务，这些任务必须按指定的过程执行。

通信内聚。指模块内的所有处理元素都在同一个数据结构上操作，或者各处理使用相同的输入数据或者产生相同的输出数据。

顺序内聚。指一个模块中的各个处理元素都密切相关于同一功能且必须顺序执行，前一功能元素的输出就是下一功能元素的输入。

功能内聚。这是最强的内聚，指模块内的所有元素共同作用完成一个功能，缺一不可。

耦合性和内聚性是模块独立性的两个定性标准，在将软件系统划分模块时，应尽量做到高内聚、低耦合，提高模块的独立性。

6.1.1.3 系统总体结构设计

系统总体结构设计是要根据系统分析的要求和组织的实际情况对新系统的总体结构形式和可利用的资源进行大致设计，这是一种宏观、总体上的设计和规划。下面介绍系统总

体设计的主要内容。

1. 系统结构设计原则

为保证总体结构设计顺利完成，应遵循以下几条原则。

(1) 分解-协调原则。整个系统是一个整体，具有整体目的和功能，但这些目的和功能的实现又是由相互联系的各个组成部分共同工作的结果。解决复杂问题的一个很重要的原则就是把它分解成多个小问题分别处理，在处理过程中根据系统总体要求协调各部门的关系。

(2) 自顶向下的原则。首先抓住系统总的功能目的，然后逐层分解，即先确定上层模块的功能，再确定下层模块的功能。

(3) 信息隐蔽、抽象的原则。上层模块只规定下层模块做什么和所属模块间的协调关系，但不规定怎么做，以保证各模块的相对独立性和内部结构的合理性，使得模块与模块之间层次分明，易于理解、实施和维护。

(4) 一致性原则。要保证整个软件设计过程中具有统一的规范、统一的标准和统一的文件模式等。

(5) 明确性原则。每个模块必须功能明确、接口明确，消除多重功能和无用接口。

(6) 模块之间的耦合尽可能小，模块的内聚度尽可能高。

(7) 模块的扇入系数和扇出系数要合理。一个模块直接调用其他模块的个数称为模块的扇出系数；反之，一个模块被其他模块调用时，直接调用它的模块个数称为模块的扇入系数。模块的扇入、扇出系数必须适当。经验表明，一个设计得好的系统的平均扇入、扇出系数通常是 3 或 4，一般不应超过 7，否则会引起出错概率的增大。但菜单调用型模块的扇入与扇出系数可以大一些，公用模块的扇入系数可以大一些。

(8) 模块的规模适当。过大的模块常常使系统分解得不充分，其内部可能包含了若干部分的功能，因此有必要进一步把原有的模块分解成若干功能尽可能单一的模块。但分解也必须适度，因为过小的模块有可能降低模块的独立性，造成系统接口的复杂性。

2. 子系统划分

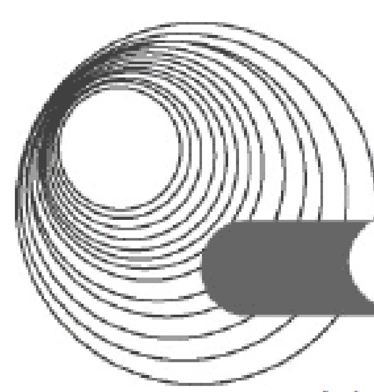
1) 子系统划分的原则

为了便于今后的系统开发和系统运行，子系统的划分应遵循以下几点原则。

(1) 子系统要具有相对独立性。子系统的划分，必须使得子系统的内部功能、信息等各方面的凝聚性较好。子系统独立可以减少子系统间的相互影响，有利于多人分工开发不同的模块，从而提高软件产品的生产率，保证软件产品的质量，同时也增强了系统的可维护性和适应性。

(2) 子系统之间数据的依赖性尽量小。子系统之间的联系要尽量减少，接口要简单明确。一个内部联系强的子系统对外部的联系必然很少，所以在划分的时候，应将联系较多者列入子系统内部，而剩余的一些分散、跨度比较大的联系，就成为这些子系统间的联系和接口。这样划分的子系统，将来调试、维护和运行都是非常方便的。

(3) 子系统划分的结果应使数据冗余较小。如果把相关的功能数据分布到各个不同的子系统中，则会有大量的原始数据需要调用，大量的中间结果需要保存和传递，大量的计算工作将要重复进行，从而使得程序结构紊乱，数据冗余，不仅给编码带来很大的困难，



而且系统的工作效率也大大降低。

(4) 子系统的设置应考虑今后管理发展的需要。子系统的设置仅依靠上述系统分析的结构是不够的，因为现存的系统由于各种原因，很可能没有考虑到一些高层次管理决策的要求。

(5) 子系统的划分应便于系统分阶段实现。信息系统的开发是一项较大的工程，它的实现一般要分批进行，所以子系统的划分应能适应这种分期分批的实施。另外，子系统的划分还必须兼顾组织结构的要求。

(6) 子系统的划分应考虑到各类资源的充分利用。一个适当的子系统划分应该既考虑有利于各种设备资源在开发过程中的搭配使用，又考虑到各类信息资源的合理分布和充分使用，以减少系统对网络资源的过分依赖，减少输入、输出和通信等设备压力。

2) 子系统结构设计

子系统结构设计的任务是确定划分后的子系统模块结构，并画出模块结构图。在进入这个过程中必须考虑以下几个问题。

(1) 每个子系统如何划分成多个模块。

(2) 如何确定子系统之间、模块之间传送的数据及其调用关系。

(3) 如何评价并改进模块结构的质量。

(4) 如何从数据流图导出模块结构图。

3. 系统模块结构设计

1) 模块的概念

模块是组成系统的基本单位，它的特点是可以组合、分解和更换。系统中的任何一个处理功能都可以看成是一个模块。根据功能具体化程度的不同，模块可以分为逻辑模块和物理模块。在系统逻辑模型中定义的处理功能可视为逻辑模块。物理模块是逻辑模块的具体化，可以是一个计算机程序、子程序或若干条程序语句，也可以是人工过程的某项具体工作。

一个模块应具备以下 4 个要素。

(1) 输入和输出。模块的输入来源和输出去向都是同一个调用者，即一个模块从调用者那里取得输入，进行加工后再把输出返回给调用者。

(2) 处理功能。指模块把输入转换成输出所做的工作。

(3) 内部数据。指仅供该模块本身引用的数据。

(4) 程序代码。指用来实现模块功能的程序。

前两个要素是模块外部特性，反映了模块的外貌。后两个要素是模块的内部特性。在结构化设计中，主要考虑的是模块的外部特性，对其内部特性只做必要了解，具体的实现将在系统实施阶段完成。

2) 模块结构图

为了保证系统设计工作的顺利进行，结构设计应遵循以下原则。

(1) 所划分的模块其内部的凝聚性要强，模块之间的联系要少，即模块具有较强的独立性。

(2) 模块之间的连接只能存在上下级之间的调用关系，不能有同级之间的横向联系。

(3) 整个系统呈树状结构，不允许网状结构或交叉调用关系出现。

(4) 所有模块(包括后继 IPO 图)都必须严格地分类编码并建立归档文件。

模块结构图主要关心的是模块的外部属性，即上下级模块、同级模块之间的数据传递和调用关系，并不关心模块的内部。

模块结构图是结构化设计中描述系统结构的图形工具。作为一种文档，它必须严格地定义模块的名字、功能和接口，同时还应当在模块结构图上反映出结构化设计的思想。

4. *数据存储设计*

信息系统的主要任务是从大量的数据中获得管理所需要的信息，这就必须存储和管理大量的数据。因此，建立一个良好的数据组织结构和数据库，使整个系统都可以迅速、方便、准确地调用和管理所需的数据，是衡量信息系统开发工作好坏的主要指标之一。

数据结构组织和数据库或文件设计，就是要根据数据的不同用途、使用要求、统计渠道和安全保密性等来决定数据的整体组织形式、表或文件的形式，以及决定数据的结构、类别、载体、组织方式、保密级别等一系列的问题。

一个好的数据结构和数据库应该充分满足组织的各级管理要求，同时还应该使后继系统的开发工作方便、快捷、系统开销(如占用空间、网络传输频度、磁盘或光盘读写次数等)小、易于管理和维护。

在建立了数据的整体结构之后，剩下的就是要确定数据的资源分布和安全保密性。其中，数据资源的分布是针对分布数据库系统而言的，而安全保密属性的定义则是针对某些特殊信息，例如财务数据等而言的。

(1) 数据资源分布。如果所规划和设计的系统是在网络环境之下，那么数据库设计必须考虑整个数据资源在网络各节点(包括网络服务器)上的分配问题。

(2) 数据的安全保密。一般数据库软件都提供定义数据安全保密性的基本功能。系统所提供的安全保密功能一般有 8 个等级(0-7 级)，4 种不同方式(只读、只写、删除、修改)，而且允许用户利用这 8 个等级的 4 种方式对每一个表自由地进行定义。

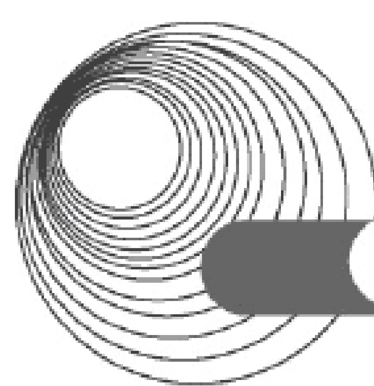
6.1.1.4　系统文档

信息系统的文档是系统建设过程的“痕迹”，是系统维护人员的指南，是开发人员与用户交流的工具。规范的文档意味着系统是按照工程化开发的，意味着信息系统的质量有了形式上的保障。文档的欠缺、文档的随意性和文档的不规范，极有可能导致原来的开发人员流动以后，系统不可维护、不可升级，变成了一个没有扩展性、没有生命力的系统。

信息系统的文档不仅包括应用软件开发过程中产生的文档，还包括硬件采购和网络设计中形成的文档：不仅包括上述有一定格式要求的规范文档，也包括系统建设过程中的各种来往文件、会议纪要、会计单据等资料形成的不规范文档，后者是建设各方谈判甚至索赔的重要依据；不仅包括系统实施记录，也包括程序资料和培训教程等。

6.1.2　典型例题分析

例　软件开发过程中，需求分析阶段的输出不包括 (19) 。(2016 年上半年上午试题 19)



(19) A．数据流图　　B．实体联系图　C．数据字典　　D．软件体系结构图

答案：D

解析：需求分析阶段的任务是确定软件系统功能，通常用数据流图、E-R 图(实体联系图)、状态转换图、数据字典描述系统的逻辑模型。

软件体系结构图是对软件系统结构的总体设计的图形显示，表示了一个系统(或功能模块)的层次分解关系，模块之间的调用关系，以及模块之间数据流和控制流信息的传递关系，它是描述系统物理结构的主要图表工具。

6.1.3　同步练习

1. 阅读下列说明和图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。(2016 年上半年下午试题一)

【说明】

某会议中心提供举办会议的场地设施和各种设备，供公司与各类组织机构租用。场地包括一个大型报告厅、一个小型报告厅以及诸多会议室。这些报告厅和会议室可提供的设备有投影仪、白板、视频播放/回放设备、计算机等。为了加强管理，该中心欲开发一套会议预订系统，系统的主要功能如下。

(1) 检查可用性。客户提交预订请求后，检查预订表，判定所申请的场地是否在申请日期内可用；如果不可用，返回不可用信息。

(2) 临时预订。会议中心管理员收到客户预定请求的通知之后，提交确认。系统生成新临时预订存入预订表，并对新客户创建一条客户信息记录加以保存。根据客户记录给客户发送临时预订确认信息和支付定金要求。

(3) 分配设施与设备。根据临时预订或变更预定的设备和设施需求，分配所需设备(均能满足用户要求)和设施，更新相应的表和预订表。

(4) 确认预订。管理员收到客户支付定金的通知后，检查确认，更新预订表，根据客户记录给客户发送预订确认信息。

(5) 变更预订。客户还可以在支付余款前提交变更预订请求，对变更的预订请求检查可用性，如果可用，分配设施和设备；如果不可用，返回不可用信息。管理员确认变更后，根据客户记录给客户发送确认信息。

(6) 要求付款。管理员从预订表中查询距预订的会议时间两周内的预定，根据客户记录给满足条件的客户发送支付余款要求。

(7) 支付余款。管理员收到客户余款支付的通知后，检查确认，更新预订表中的已支付余款信息。

现采用结构化方法对会议预定系统进行分析与设计，获得如图 6-1 所示的上下文数据流图和图 6-2 所示的 0 层数据流图(不完整)。



图 6-1　上下文数据流图



图 6-2　0 层数据流图

【问题 1】(2 分)

使用说明中的词语，给出图 6-1 中的实体 E1～E2 的名称。

【问题 2】(4 分)

使用说明中的词语，给出图 6-2 中的数据存储 D1～D4 的名称。

【问题 3】(6 分)

根据说明和图中术语，补充图 6-2 之中缺失的数据流及其起点和终点。

【问题 4】(3 分)

如果发送给客户的确认信息是通过 Email 系统向客户信息中的电子邮件地址进行发送

的，那么需要对图 6-1 和图 6-2 进行哪些修改？用 150 字以内文字加以说明。

2. 阅读下列说明和图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。(2015 年下半年下午试题一)

【说明】

某慕课教育平台欲添加在线作业批改系统，以实现高效的作业提交与批改，并进行统计。学生和讲师的基本信息已经初始化为数据库中的学生表和讲师表。系统的主要功能如下：

(1) 提交作业。验证学生标识后，学生将电子作业通过在线的方式提交，并进行存储。系统给学生发送通知表明提交成功，通知中包含唯一编号；并通知讲师有作业提交。

(2) 下载未批改作业。验证讲师标识后，讲师从系统中下载学生提交的作业。下载的作业将显示在屏幕上。

(3) 批改作业。讲师按格式为每个题目进行批改打分，并进行整体评价。

(4) 上传批改后的作业。将批改后的作业(包括分数和评价)返回给系统，进行存储。

(5) 记录分数和评价。将批改后的作业的分数和评价记录在学生信息中，并通知学生作业已批过。

(6) 获取已批改作业。根据学生标识，给学生查看批改后的作业，包括提交的作业、分数和评价。

(7) 作业抽检。根据教务人员标识抽取批改后的作业样本，给出抽检意见，然后形成抽检报告给讲师。

现采用结构化方法对在线作业批改系统进行分析与设计，获得如图 6-3 所示的上下文数据流图和图 6-4 所示的 0 层数据流图。

【问题 1】(3 分)

使用说明中的词语，给出图 6-3 中的实体 E1～E3 的名称。

【问题 2】(4 分)

使用说明中的词语，给出图 6-4 中的数据存储 D1～D4 的名称。

【问题 3】(6 分)

根据说明和图中术语，补充图 6-4 中缺失的数据流及其起点和终点。

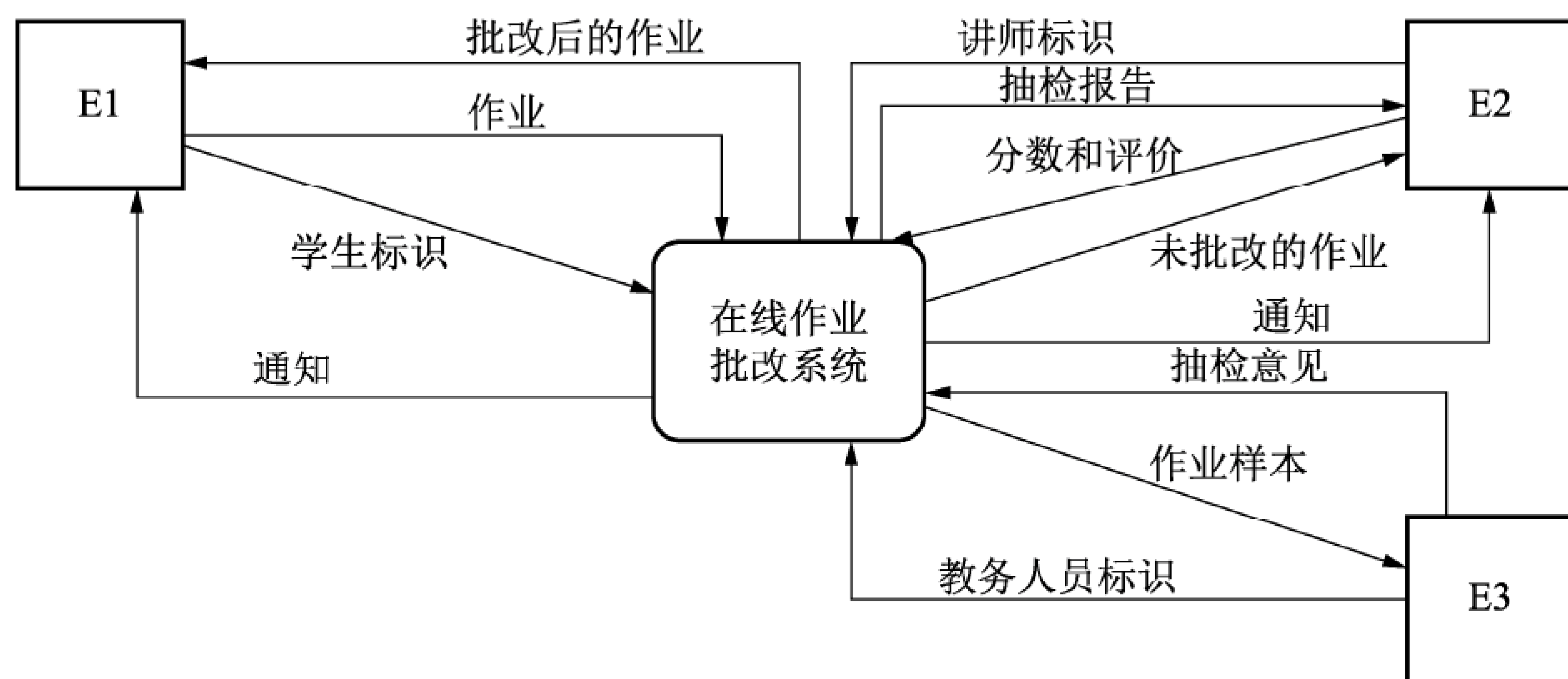


图 6-3 上下文数据流图

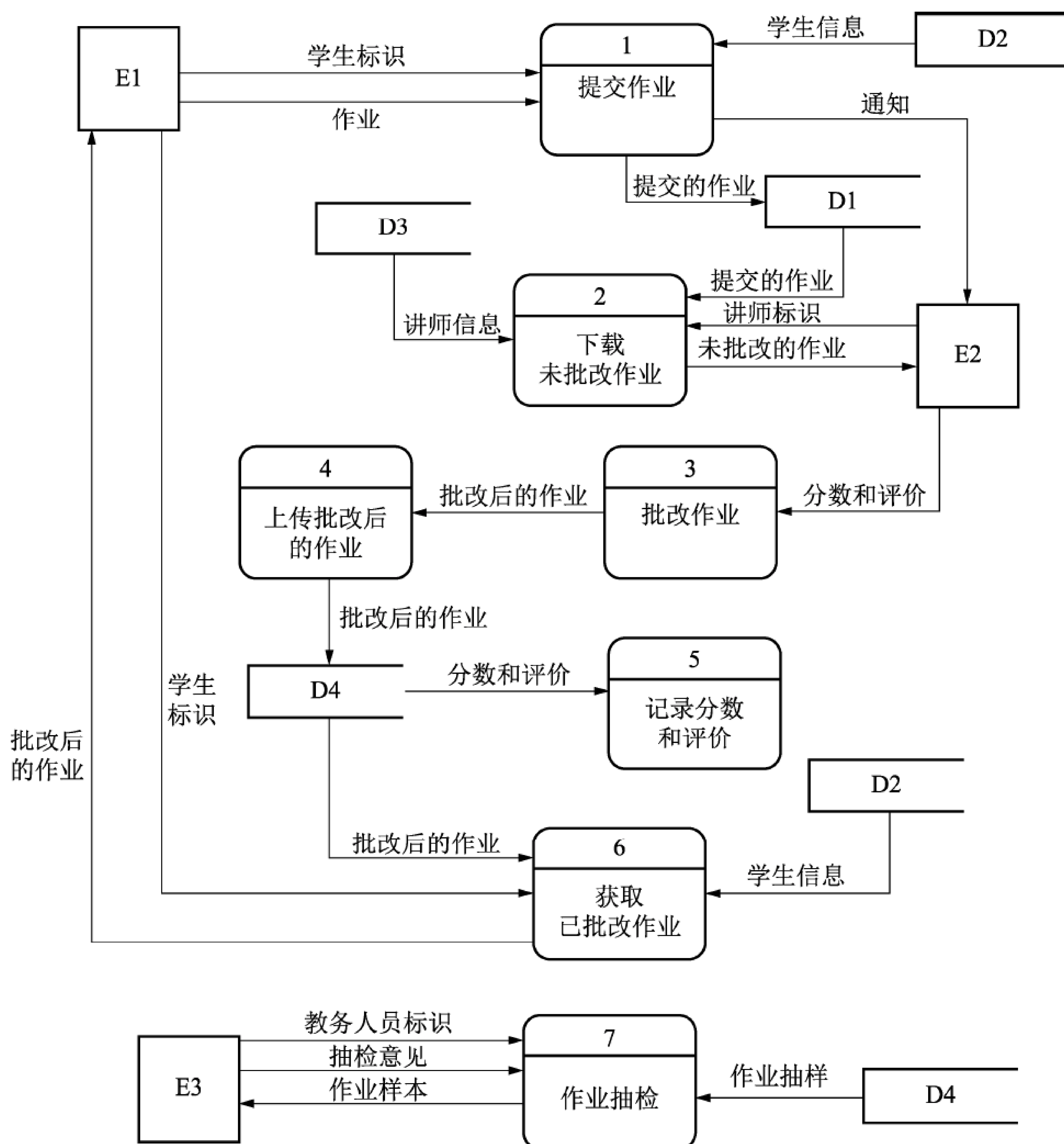


图 6-4　0 层数据流图

【问题 4】(2 分)

若发送给学生和讲师的通知是通过第三方 Email 系统进行的，则需要对图 6-3 和图 6-4 进行哪些修改？用 100 字以内文字加以说明。

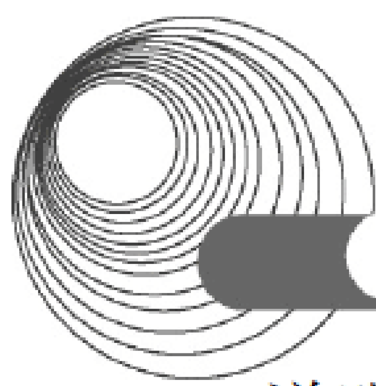
3. 阅读下列说明和图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。(2015 年上半年下午试题一)

【说明】

某大学为进一步推进无纸化考试，欲开发一套考试系统。系统管理员能够创建包括专业方向、课程编号、任课教师等相关考试基础信息，教师和学生进行考试相关的工作。系统与考试有关的主要功能如下。

(1) 考试设置。教师制定试题(题目和答案)，制定考试说明、考试时间和提醒时间等考试信息，录入参加考试的学生信息，并分别进行存储。

(2) 显示并接收解答。根据教师设定的考试信息，在考试有效时间内向学生显示考试



说明和题目，根据设定的考试提醒时间进行提醒，并接收学生的解答。

(3) 处理解答。根据答案对接收到的解答数据进行处理，然后将解答结果进行存储。

(4) 生成成绩报告。根据解答结果生成学生个人成绩报告，供学生查看。

(5) 生成成绩单。对解答结果进行核算后生成课程成绩单供教师查看。

(6) 发送通知。根据成绩报告数据，创建通知数据并将通知发送给学生；根据成绩单数据，创建通知数据并将通知发送给教师。

现采用结构化方法对考试系统进行分析与设计，获得如图 6-5 所示的上下文数据流图和图 6-6 所示的 0 层数据流图。

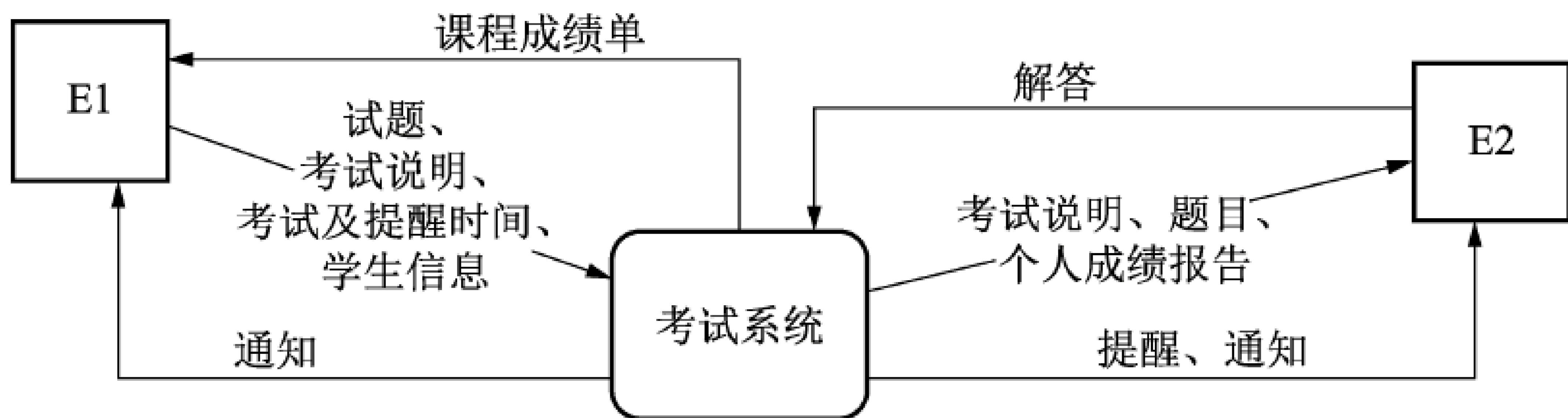


图 6-5 上下文数据流图

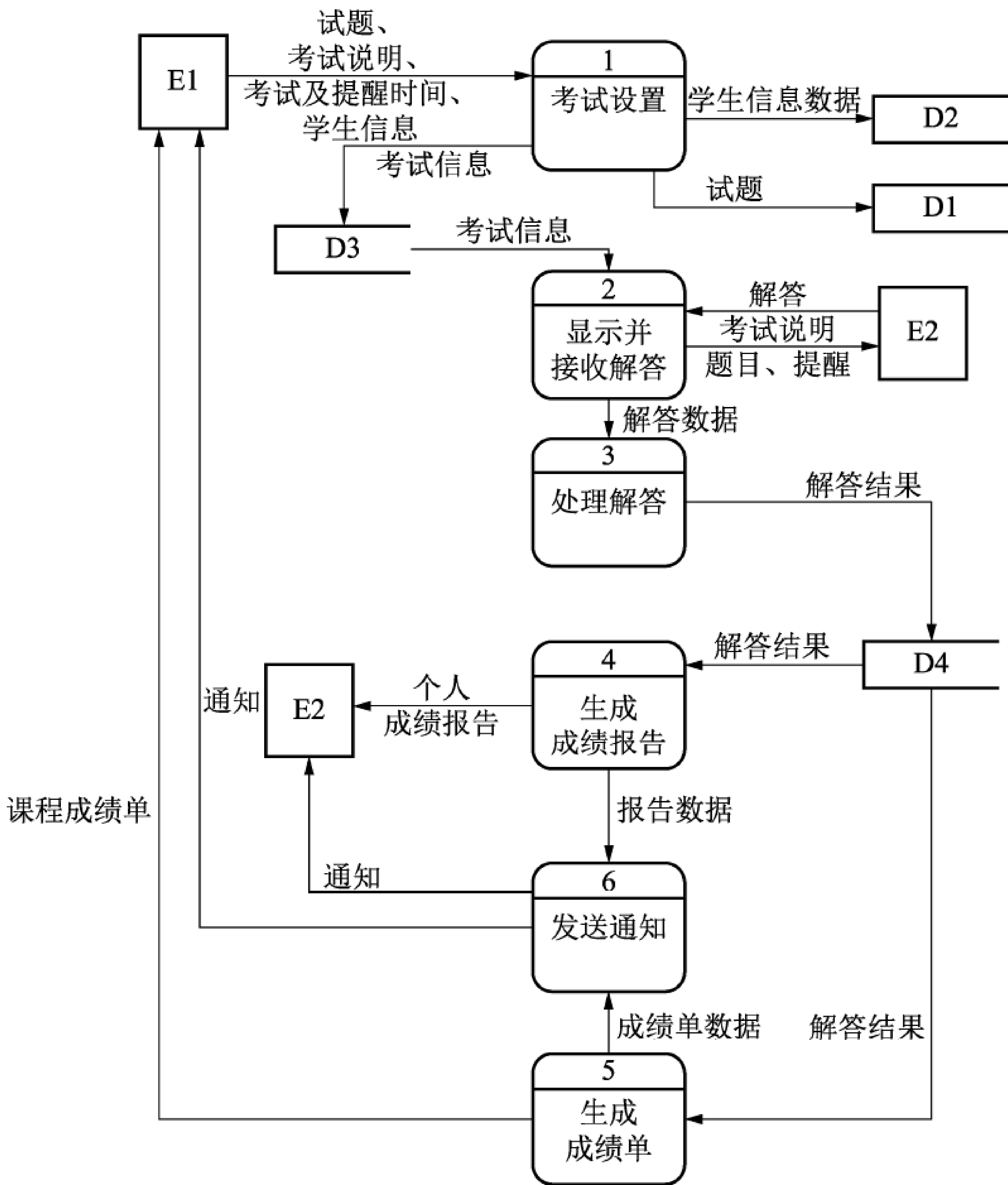


图 6-6 0 层数据流图

【问题 1】(2 分)

使用说明中的词语，给出图 6-5 中的实体 E1～E2 的名称。

【问题 2】(4 分)

使用说明中的词语，给出图 6-6 中的数据存储 D1～D4 的名称。

【问题 3】(4 分)

根据说明和图中词语，补充图 6-6 中缺失的数据流及其起点和终点。

【问题 4】(5 分)

图 6-6 所示的数据流图中，功能(6)发送通知包含创建通知并发送给学生或老师。请分解图 6-6 中加工(6)，将分解出的加工和数据流填入答题纸的对应栏内。(注：数据流的起点和终点须使用加工的名称描述)

4. 阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。(2014 年下半年下午试题一)

【说明】

某大型披萨加工和销售商为了有效管理生产和销售情况，欲开发一套披萨信息系统，其主要功能扣下：

(1) 销售。处理客户的订单信息，生成销售订单，并将其记录在销售订单表中。销售订单记录了订购者、所订购的披萨、期望的交付日期等信息。

(2) 生产控制。根据销售订单以及库存的披萨数量，制定披萨生产计划(包括生产哪些披萨、生产顺序和生产量等)，并将其保存在生产计划表中。

(3) 生产。根据生产计划和配方表中的披萨配方，向库存发出原材料申领单，将制作好的披萨的信息存入库存表中，以便及时进行交付。

(4) 采购。根据所需原材料及库存量，确定采购数量，向供应商发送采购订单，并将其记录在采购订单表中；得到供应商的供应量，将原材料数量记录在库存表中，在采购订单表中标记已完成采购的订单。

(5) 运送。根据销售订单将披萨交付给客户，并记录在交付记录表中。

(6) 财务管理。在披萨交付后，为客户开具费用清单，收款并出具收据；依据完成的采购订单给供应商支付原材料费用并出具支付细节；将收款和支付记录存入收支记录表中。

(7) 存储。检查库存的原材料、披萨和未完成订单，确定所需原材料。

现采用结构化方法对披萨信息系统进行分析与设计，获得如图 6-7 所示的上下文数据流图和图 6-8 所示的 0 层数据流图。

【问题 1】(4 分)

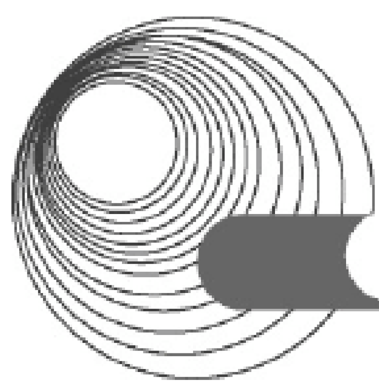
根据说明中的词语，给出图 6-7 中的实体 E1~E2 的名称。

【问题 2】(5 分)

根据说明中的词语，给出图 6-8 中的数据存储 D1~D5 的名称。

【问题 3】(6 分)

根据说明中的词语，补充图 6-8 中缺失的数据流及其起点和终点。



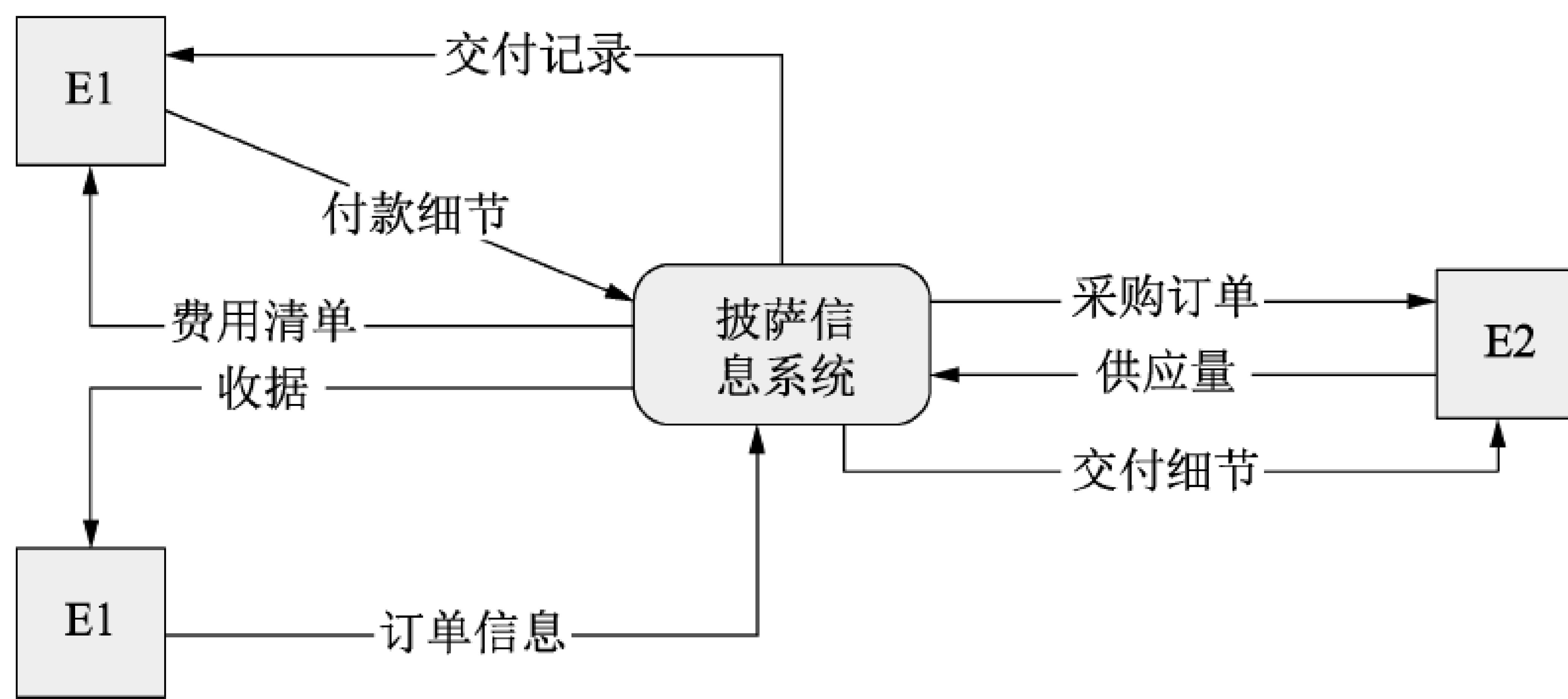


图 6-7　上下文数据流图

图 6-8　0 层数据流图

6.1.4　同步练习参考答案

试题 1

答案：

【问题 1】

E1：客户

E2：管理员

【问题 2】

D1：预定表

D2：客户信息记录表

D3：设施表

D3：设备表

【问题 3】

预订确认信息 起点：4 确认预定终点：E1

客户信息 起点：E1 终点 2 临时预定

【问题 4】

图 6-1 中：增加外部实体“第三方 Email 系统”，将临时预订/预订/变更确认信息终点均修改至“第三方 Email 系统”。

图 6-2 中：增加外部实体“第三方 Email 系统”，增加加工“发送邮件”，将临时预订/预订/变更确认信息终点均修改至“发送邮件”加工，并增加从 D2 到“发送邮件”加工的数据流“电子邮件地址”，再从发送邮件加工引出数据流 临时预订/预订/变更确认信息 终点为 第三方 Email 系统。

解析：

该题以会议预订系统来考查学生对数据流图知识点的掌握程度。从题目的问答形式上来看和往年相似。

【问题 1】

根据 0 层数据流中 E1 向系统发送预订请求数据流可知，E1 实体为“客户”；从预订请求通知到临时预订确认可知 E2 实体为“管理员”。

【问题 2】

根据题目对功能的描述，结合 0 层数据流图，新临时预订提交、变更的预定请求提交等，可知 D1 为预定表；新客户信息存入 D2 中，可知 D2 为客户信息记录表；根据分配设施和设备数据流，可以得到 D3、D4 分别为设施表和设备表。

【问题 3】

由“确认预定”收到客户支付定金的通知后，检查确认更新预订表，同时要向客户发送预订确认信息，存在一个起点为 4 确定预订点 到 终点为 E1 的数据流，即预订确认信息数据流；根据临时预订描述，首先要有客户发送预订请求，提交确认，系统生成新临时预订存入预订表，所以存在一个起点为客户即 E1，终点为 2 临时预订的数据流，即客户临时预订信息数据流。

【问题 4】

略

试题 2

参考答案：

【问题 1】

E1：学生　E2：讲师　E3：教务人员

【问题 2】

D1：作业　D2：学生表　D3：讲师表　D4：批改后的作业

【问题 3】

数据流名称：通知　起点：提交作业　终点：E1

数据流名称：抽检报告　起点：作业抽检　终点：E2

数据流名称：分数和评价　起点：记录分数和评价　终点：D2

数据流名称：通知　起点：记录分数和评价　终点：E1

【问题 4】

增加外部实体“第三方 Email 系统”，将原来的两条“通知”数据流合并为一条“通知”数据流，终点为“第三方 Email 系统”。

解析：

【问题 1】由系统功能说明知，学生将电子作业通过在线的方式提交系统，所以 E1 实体为“学生”；讲师从系统中下载学生提交的作业，进行批改打分并做整体评价，所以 E2 实体为“讲师”；教务人员抽取批改后的作业样本，给出抽检意见，所以 E2 实体为“教务人员”。

【问题 2】学生和讲师的基本信息已经初始化为数据库中的学生表和讲师表，所以数据存储 D2、D3 的名称分别为“学生表”和“讲师表”。学生将电子作业完成后通过在线的方式提交并存储，因此数据存储 D1 存储的是学生的作业，名称为“作业”；讲师批改完学生作业后，需将批改后的作业(包括分数和评价)返回给系统并进行存储，所以数据存储 D4 的名称为“批改后的作业”。

【问题 3】学生提交作业之后，系统要给学生发送通知消息，显然数据流图中缺少这样一条数据流；讲师批改完作业后，将批改后的作业的分数和评价记录在学生信息，并通知学生作业批改过，这里缺少两条数据流；最后教务人员抽检后将结果报告讲师，这条数据流这里缺少。

【问题 4】略。

试题 3

参考答案：

【问题 1】

E1：教师　E2:学生

【问题 2】

D1：试题　D2：学生信息　D3：考试信息　D4：解答结果

【问题 3】

数据流名称：题目，起点：D1，终点：2 显示并接收解答。

数据流名称：答案，起点：D1，终点：3 处理解答。

【问题 4】

分解为：创建通知数据，发送通知给学生或老师

数据流名称：通知数据；起点：创建通知数据；终点：发送通知给学生或老师。

解析：

【问题 1】根据题目说明，教师指定试题、考试说明、考试时间和考试题型，并录入学生信息，系统将考试说明、题目显示给学生，由学生解答。显然 E1 实体为“教师”，E2 实体为“学生”。

【问题 2】教师将考试信息、实体、学生信息录入系统，根据数据流，显然 D1 为“试题”，D2 为“学生”信息，D3 为“考试信息”。学生解答后，由系统处理解答结果，并保存学生的解答结果，同时根据解答结果生成学生个人成绩报告，显然 D4 为“解答结果”。

【问题 3】系统要向学生显示考试说明和题目，显然，缺少试题到加工 2 的数据流。学生做完题后，系统要根据答案对接收到的解答数据进行处理，这里缺少答案到处理 3 的数据流，根据题设知，答案包含在老师指定的试题中。

【问题 4】根据题设，系统要根据成绩报告数据，创建通知数据并将通知发送给学生；根据成绩单数据，创建通知数据并将通知发送给教师。

试题 4

答案：

【问题 1】

E1：客户；E2：供应商

【问题 2】

D1：销售订单表；D2：库存表；D3：生产计划表；D4：配方表；D5：采购订单表

【问题 3】

(1) 数据流名称：支付细节；起点：财务管理；终点：E2

(2) 数据流名称：销售订单：起点：销售订单表；终点：5 运送

(3) 数据流名称：生产计划：起点：D3；终点：3 生产

(4) 数据流名称：库存量；起点：D2；终点：4 采购

(5) 数据流名称：原材料数量：起点：4 采购；终点：D2

(6) 数据流名称：未完成订单：起点：销售订单表；终点：7 存储。

解析：

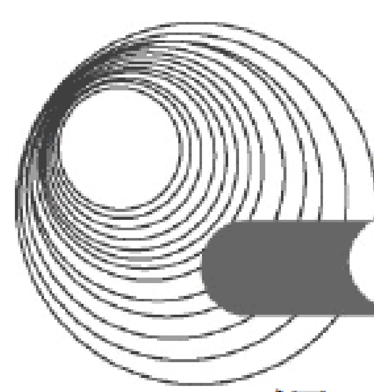
该题以披萨信息系统为载体来考查学生对数据流图知识点的掌握程度。从题目的问答形式上来看，和往年相似，要求补充外部实体、补充缺失数据流、找出外部存储。

【问题 1】

根据 0 层数据流中财务管理为客户开具费用清单数据流可知，E1 实体为客户；从向供应商发送采购订单、得到供应商的供应量可知，E2 实体为供应商。

【问题 2】

根据题中对功能的描述，以及 0 层数据流：生产控制，制订生产计划，并将其保存在生产计划表中可知，D3 为生产计划表；生产，将制作好的披萨的信息存入库存表中，可知 D2 为库存表；采购，在采购订单表中标记已完成的订单可知 D5 为采购订单表；生产，由“根据生产计划和配方表中的披萨配方，向库存发出原料申请”可知 D4 为配方表；根



据“处理客户的订单信息，生成销售订单。并将其记录在销售订单表中”可知 D1 为销售订单表。

【问题 3】

本题考查补充分层 DFD 中的数据流。在分层 DFD 中，需要保持父图和子图的平衡，即父图中某加工的输入输出数据流必须与其子图的输入输出数据流在数量和名字上相同，或者父图的一个输入(或输出)数据流对应于子图中几个输入(或输出)数据流，而子图中组成这些数据流的数据项全体正好是附图中的一个数据流。

由“财务管理”段中“依据完成的采购订单给供应商支付原材料费用并出具支付细节”的描述，存在一个起点为财务管理，重点为供应商即 E2 的数据流，即支付细节数据流；由“运送”段中“根据销售订单将披萨交付给客户，并记录在交付记录表中”可知存在一个由“销售订单”指向“运送”的数据流，即销售订单数据流；由“采购”段中“根据所需原材料及库存量，确定采购数量”可知存在由“库存表”指向“采购”的数据流，数据流为库存量；根据“存储”段中“检查库存的原材料、披萨和未完成订单，确定所需原料”可知，存在一个由“订单表”指向“存储”的数据流，数据流为未完成的订单。

6.2　结构化分析方法

6.2.1　考点辅导

结构化分析与设计方法是一种面向数据流的传统软件开发方法，它以数据流为中心构建软件的分析模型和设计模型。结构化分析(Structured Analysis，SA)、结构化设计(Structured Design，SD)和结构化程序设计(Structured Programming Design，SPD)构成了完整的结构化方法。

1. 结构化分析方法概述

抽象和分解是处理任何复杂问题的两个基本手段。

抽象是指忽略一个问题中与当前目标无关的那些方面，以便更充分地关注与当前目标有关的方面。对于一个复杂的问题，人们很难一下子考虑问题的所有方面和全部细节，通常可以把一个大问题分解成若干个小问题，将每个小问题再分解成若干个更小的问题，经过多次逐层分解，每个最底层的问题都是足够简单、容易解决的，于是复杂的问题也就迎刃而解了。这个过程就是分解的过程。

结构化方法就是采用这种自顶向下逐层分解的思想进行分析建模的。自顶向下逐层分解充分体现了分解和抽象的原则。随着分解层次的增加，抽象的级别也越来越低，即越来越接近问题的解。自顶向下的过程是分解的过程，自底向上的过程是抽象的过程。

结构化方法的分析结果由以下几部分组成：一套分层的数据流图、一本数据词典、一组小说明(也称加工逻辑说明)、补充材料。

2. 数据流图

数据流图也称数据流程图(Data Flow Diagram，DFD)，它是一种便于用户理解、分析系

统数据流程的图形工具。它摆脱了系统的物理内容，精确地在逻辑上描述系统的功能、输入、输出和数据存储等，是系统逻辑模型的重要组成部分。

1)　数据流图的基本图形元素

数据流图中的基本图形元素包括数据流(Data Flow)、加工(Process)、数据存储(Data Store)和外部实体(Extemal Agent)。其中，数据流、加工和数据存储用于构建软件系统内部的数据处理模型；外部实体表示存在于系统之外的对象，用来帮助用户理解系统数据的来源和去向。

(1)　数据流。

数据流由一组固定成分的数据组成，表示数据的流向。在 DFD 中，数据流的流向可以有以下几种：从一个加工流向另一个加工；从加工流向数据存储(写)；从数据存储流向加工(读)；从外部实体流向加工(输入)；从加工流向外部实体(输出)。

DFD 中的每个数据流用一个定义明确的名字表示。除了流向数据存储或从数据存储流出的数据流不必命名外，每个数据流都必须有一个合适的名字，以反映该数据流的含义。

值得注意的是，DFD 中描述的是数据流，而不是控制流。

数据流或者由具体的数据属性(也称为数据结构)构成，或者由其他数据流构成。组合数据流是由其他数据流构成的数据流，它们用于在高层的数据流图中组合相似的数据流，以使数据流图更便于阅读。

(2)　加工。

加工描述了输入数据流到输出数据流之间的变换，也就是输入数据流经过什么处理后变成了输出数据流。每个加工都有一个名字和编号。编号能反映出该加工位于分层 DFD 中的哪个层次和哪张图中，也能够看出它是哪个加工分解出来的子加工。

一个加工可以有多个输入数据流和多个输出数据流，但至少有一个输入数据流和一个输出数据流。

(3)　数据存储。

数据存储用来存储数据。通常，一个流入加工的数据流经过加工处理后就消失了，而它的某些数据(或全部数据)可能被加工成输出数据流，流向其他加工或外部实体。除此之外，在软件系统中还常常要把某些信息保存下来以供以后使用，这时可以使用数据存储。

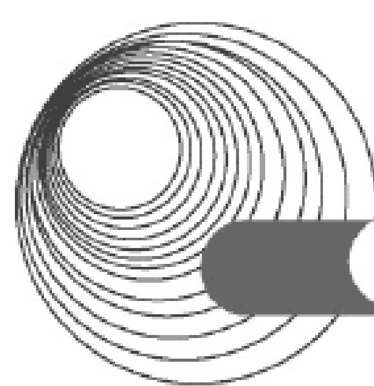
每个数据存储都有一个定义明确的名字标识。可以有数据流流入数据存储，表示数据的写入操作；也可以有数据流从数据向数据存储，表示对数据的修改。

这里要说明的是，DFD 中的数据存储在具体实现时可以用文件系统实现，也可以用数据库系统实现。数据存储的存储介质可以是磁盘、磁带或其他存储介质。

(4)　外部实体(外部主体)。

外部实体是指存在于软件系统之外的人员或组织，它指出系统所需数据的发源地(源)和系统所产生的数据的归宿地(宿)。例如，对于一个考务处理系统而言，考生向系统提供报名单(输入数据流)，所以考生是考务处理系统的一个源；而考务处理系统要将考试成绩的统计分析表(输出数据流)传递给考试中心，所以考试中心是该系统的一个宿。

在许多系统中，某个源和某个宿可以是同一个人员或组织，此时，在 DFD 中可以用同一个符号表示。考生向系统提供报名单，而系统向考生送出准考证，所以在考务处理系统中，考生既是源又是宿。



源和宿采用相同的图形符号表示，当数据流从该符号流出时，表示它是源；当数据流流向该符号时，表示它是宿；当两者皆有时，表示它既是源又是宿。

2) 数据流图的扩充符号

在 DFD 中，一个加工可以有多个输入数据流和多个输出数据流，此时可以加上一些扩充符号来描述多个数据流之间的关系。

(1) 星号(*)。

星号表示数据流之间存在“与”关系。如果是输入流则表示所有输入数据流全部到达后才能进行加工处理；如果是输出流则表示加工结束将同时产生所有的输出数据流。

(2) 加号(+)。

加号表示数据流之间存在“或”关系。如果是输入流则表示其中任何一个输入数据流到达后就能进行加工处理；如果是输入流则表示加工处理的结果是至少产生其中一个输出数据流。

(3) 异或(⊕)。

异或表示数据流之间存在“互斥”关系。如果是输入流则表示当且仅当其中一个输入流到达后才能进行加工处理；如果是输出流则表示加工处理的结果是仅产生这些输出数据流中的一个。

3) 数据流图的层次结构

从原理上讲，只要纸足够大，一个软件系统的分析模型就可以画在一张纸上。然而，一个复杂的软件系统可能涉及上百个加工和上百个数据流，甚至更多。如果将它们画在一张图上，则会十分复杂，不易阅读，也不易理解。

根据自顶向下逐层分解的思想，可以将数据流图按照层次结构来绘制，每张图中的加工个数可大致控制在“7 加减 2”的范围内，从而构成一套分层数据流图。

(1) 层次结构。

分层数据流图的顶层只有一张图，其中只有一个加工，代表整个软件系统，该加工描述了软件系统与外界之间的数据流，称为顶层图。

顶层图中的加工(即系统)经分解后的图称为 0 层图，也只有一张。处于分层数据流图最底层的图称为底层图，在底层图中，所有的加工不再进行分解。分层数据流图中的其他图称为中间层，其中至少有一个加工(也可以是所有加工)被分解成一张子图。在整套分层数据流图中，凡是不再分解成子图的加工称为基本加工。

(2) 图和加工的编号。

首先介绍父图和子图的概念。

如果某图(记为 A)中的某一个加工分解成一张子图(记为 B)，则称 A 是 B 的父图，B 是 A 的子图。若父图中有 n 个加工，则它可以有 0 一刀张子图，但每张子图只对应一张父图。

为了方便对图进行管理和查找，可以采用下列方式对 DFD 中的图和加工编号。

① 顶层图中只有一个加工(代表整个软件系统)，该加工不必编号。

② 0 层图中的加工编号分别为 1、2、3--。

③ 子图号就是父图中被分解的加工号。

④ 对于子图中加工的编号，若父图中的加工号为 X 的加工分解成某一子图，则该子图中的加工编号分别为 x.1、x.2、X.3…。

4)　分层数据流图的审查

在分层数据流图画好后，应该认真检查图中是否存在错误或不合理(不理想)的部分。

(1)　分层数据流图的一致性和完整性。

①　分层数据流图的一致性。

a. 父图与子图的平衡。

b. 数据守恒。

c. 局部数据存储。

d. 一个加工的输出数据流不能与该加工的输入数据流同名。

②　分层数据流图的完整性。

a. 每个加工至少有一个输入数据流和一个输出数据流。

b. 在整套分层数据流图中，每个数据存储应至少有一个加工对其进行读操作，另一个加工对其进行写操作。

c. 分层数据流图中的每个数据流和文件都必须命名(除了流入或流出数据存储的数据流)，并保持与数据字典一致。

d. 分层数据流图中的每个基本加工都应有一个加工规约。

(2)　构造分层 DFD 时需要注意的问题。

①　适当命名。

a. 名字应反映整个对象(如数据流、加工)，而不是只反映它的某一部分。

b. 避免使用空洞的、含义不清的名字，如“数据”“信息”“处理”“统计”等。

c. 如果发现某个数据流或加工难以命名，往往是 DFD 分解不当的征兆，此时应考虑重新分解。

②　画数据流而不是控制流。

③　避免一个加工有过多的数据流。

a. 把需要重新分解的某张图的所有子图连接成一张图。

b. 把连接后的图重新划分成几个部分，使各部分之间的联系最小。

c. 重新定义父图，即第 b 步中的每个部分作为父图中的一个加工。

d. 重新建立各子图，即第 b 步中的每个部分都是一张子图。

e. 为所有的加工重新命名并编号。

④　分解尽可能均匀。

⑤　先考虑确定状态，忽略琐碎的细节。

⑥　随时准备重画。

(3)　分解的程度。

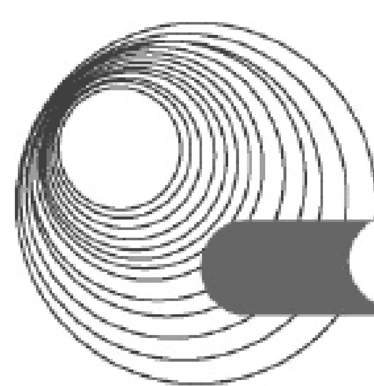
在自顶向下画数据流图时，为了便于对分解层数进行把握，可以参照以下几条与分解有关的原则。

①　7 加减 2。

②　分解应自然，概念上应合理、清晰。

③　只要不影响 DFD 的易理解性，可适当增加子加工数量，以减少层数。

④　一般来说，上层分解得快一些(即多分解几个加工)，下层分解得慢一些(即少分解几个加工)。



⑤ 分解要均匀。

3. 数据字典(DD)

数据流图描述了系统的分解，但没有对图中各成分进行说明。数据字典就是为数据流图中的每个数据流、文件、加工，以及组成数据流或文件的数据项做出说明。其中，对加工的描述称为“小说明”，也可以称为“加工逻辑说明”。

1) 数据字典的内容

数据字典有以下 4 类条目：数据流、数据项、数据存储和基本加工。数据项是组成数据流和数据存储的最小元素。源点、终点不在系统之内，故一般不在字典中说明。

(1) 数据流条目。数据流条目给出了 DFD 中数据流的定义，通常列出该数据流的各组成数据项。

(2) 数据存储条目。数据存储条目是对数据存储的定义。

(3) 数据项条目。数据项条目是不可再分解的数据单位。

(4) 基本加工条目。加工条目是用来说明 DFD 中基本加工的处理逻辑的，由于下层的基本加工是由上层的加工分解而来，只要有了基本加工的说明，就可理解其他加工。

2) 数据词典管理

词典管理主要是把词典条目按照某种格式组织后存储在词典中，并提供排序、查找和统计等功能。如果数据流条目包含了来源和去向，文件条目包含了读文件和写文件，还可以检查数据词典与数据流图的一致性。

3) 加工逻辑的描述

加工逻辑也称为“小说明”。常用的加工逻辑描述方法有结构化语言、判定表和判定树 3 种。

(1) 结构化语言。

结构化语言(如结构化英语)是一种介于自然语言和形式化语言之间的半形式化语言，是自然语言的一个受限子集。

结构化语言没有严格的语法，它的结构通常可分为内层和外层。外层有严格的语法，内层的语法比较灵活，可以接近于自然语言的描述。

① 外层。用来描述控制结构，采用顺序、选择和重复 3 种基本结构。

a. 顺序结构。一组祈使语句、选择语句、重复语句的顺序排列。祈使语句是指至少包含一个动词及一个名词，指出要执行的动作及接受动作的对象。

b. 选择结构。一般用 IF-THEN-ELSE-ENDIF、CASE-OF-ENDCASE 等关键词。

c. 重复结构。一般用 DO-WHILE-ENDDO、REPEAT-UNTIL 等关键词。

② 内层。一般采用祈使语句的自然语言短语，使用数据字典中的名词和有限的自定义词，其动词含义要具体，尽量不用形容词和副词来修饰，还可使用一些简单的算法运算和逻辑运算符号。

(2) 判定表。

在有些情况下，数据流图中某个加工的一组动作依赖于多个逻辑条件的取值。这时，用自然语言或结构化语言都不易于清楚地描述出来，而用判定表能够清楚地表示复杂的条件组合与应做的动作之间的对应关系。

判定表由 4 个部分组成，用双线分割成 4 个区域，如图 6-9 所示。

条件定义	条件取值的组合
动作定义	在各种取值的组合下应执行的动作

图 6-9　判定表结构

(3) 判定树。

判定树是判定表的变形，一般情况下它比判定表更直观，且易于理解和使用。

6.2.2　典型例题分析

例　在进行子系统结构设计时，需要确定划分后的子系统模块结构，并画出模块结构图。该过程不需要考虑 (32) 。(2015 年下半年试题 32)

(32) A．每个子系统如何划分成多个模块

B．每个子系统采用何种数据结构和核心算法

C．如何确定子系统之间、模块之间传送的数据及其调用关系

D．如何评价并改进模块结构的质量

解析：子系统结构设计的任务是确定划分后的子系统的模块结构，并画出模块结构图，这个过程中必须考虑以下几个问题。

(1) 每个子系统如何划分成多个模块。

(2) 如何确定子系统之间、模块之间传送的数据及其调用关系。

(3) 如何评价并改进模块结构的质量。

(4) 如何从数据流图导出模块结构图。

答案：B

6.2.3　同步练习

1. ______是一种面向数据流的开发方法，其基本思想是软件功能的分解和抽象。

A．结构化开发方法　　B．Jackson 系统开发方法

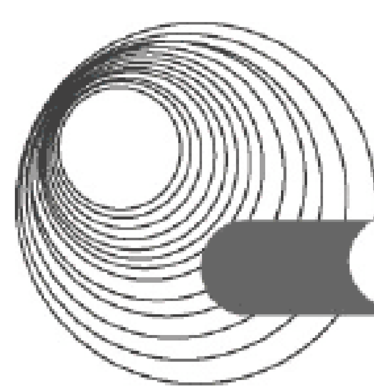
C．Booch 方法　　D．UML(统一建模语言)

2. 结构化开发方法中，数据流图是______阶段产生的成果。

A．需求分析　　B．总体设计　　C．详细设计　　D．程序编码

6.2.4　同步练习参考答案

1．A　　2．A



6.3 结构化设计方法

6.3.1 考点辅导

结构化设计(Structured Design，SD)方法是一种面向数据流的设计方法，它可以与 SA 方法衔接。结构化设计方法的基本思想是将系统设计成由相对独立、功能单一的模块组成的结构。

结构化设计方法中用结构图(Structure Chart)来描述软件系统的体系结构，指出一个软件系统由哪些模块组成，以及模块之间的调用关系。

6.3.1.1 结构化设计的步骤

结构化设计大致可以分为两步进行，第一步是建立一个满足软件需求规约的初始结构图，第二步是对结构图进行改进。

1. 建立初始结构图

结构化方法本质上是一种功能分解方法。在结构化设计时，可以将整个软件看作一个大的功能模块(结构图中的模块)，通过功能分解将其分解成若干个较小的功能模块，每个较小的功能模块还可以进一步分解，直到得到一组不必再分解的模块(结构图中的底层模块)。当一个功能模块分解成若干个子功能模块时，该功能模块实际上就是根据业务流程调用相应的子功能模块，并根据其功能要求对子功能的结果进行处理，最终实现其功能要求。

功能模块的分解应满足自顶向下、逐步求精、信息隐蔽、高内聚低耦合等设计准则，模块的大小应适中。通常，一个模块的大小以 so～ioo 行程序代码为宜，即一个模块的程序代码可以写在 1-2 页纸上。

2. 对结构图的改进

初始结构图往往存在一些不合理的设计(包括不合理的模块分解)，因此，可根据设计准则对其进行改进。

3. 书写设计文档

在概要设计完成之后应书写设计规格说明，特别要为每个模块书写模块的功能、接口、约束和限制等，必要时可建立模块开发卷宗。

4. 设计评审

对设计结果及文档进行评审。

6.3.1.2 数据流图到软件体系结构的映射

结构化设计是将结构化分析的结果(数据流图)映射成软件的体系结构(结构图)。根据信息流的特点，可将数据流图分为变换型数据流图和事务型数据流图，其对应的映射分别称为变换分析和事务分析。

1. 信息流的类型

在需求分析阶段，用 SA 方法产生了数据流图。面向数据流的设计能方便地将 DFD 转换成程序结构图。DFD 中从系统的输入数据流到系统的输出数据流的一连串连续变换形成了一条信息流。DFD 的信息流大体上可以分为两种类型：变换流和事务流。

(1) 变换流。信息沿着输入通路进入系统，同时将信息的外部形式转换成内部表示，然后通过变换中心(也称主加工)处理，再沿着输出通路转换成外部形式离开系统。具有这种特性的信息流称为变换流。变换流型的 DFD 可以明显地分成输入、变换(主加工)和输出三大部分。

(2) 事务流。信息沿着输入通路到达一个事务中心，事务中心根据输入信息(即事务)的类型在若干个动作序列(称为活动流)中选择一个来执行，这种信息流称为事务流。事务流有明显的事务中心，各活动流以事务中心为起点呈辐射状流出。

2. 变换分析

变换分析是从变换流型的 DFD 导出程序结构图。

1) 确定输入流和输出流，分离出变换中心

把 DFD 中系统输入端的数据流称为物理输入，系统输出端的数据流称为物理输出。物理输入通常要经过编辑、格式转换、合法性检查、预处理等辅助性的加工才能为主加工的真正输入(称为逻辑输入)。从物理输入端开始，一步步向系统的中间移动，可找到离物理输入端最远，但仍可被看作系统输入的那个数据流，这个数据流就是逻辑输入。同样，由主加工产生的输出(称为逻辑输出)通常也要经过编辑、格式转换、组成物理块、缓冲处理等辅助加工才能变成物理输出。从物理输出端开始，一步步向系统的中间移动，可找到离物理输出端最远，但仍可被看作系统输出的那个数据流，这个数据流就是逻辑输出。

DFD 中从物理输入到逻辑输入的部分构成系统的输入流，从逻辑输出到物理输出的部分构成系统的输出流，位于输入流和输出流之间的部分就是变换中心。

2) 第一级分解

第一级分解主要是设计模块结构的顶层和第一层。一个变换流型的 DFD 可以映射成如图 6-10 所示的程序结构图。图中顶层模块的功能就是整个系统的功能。输入控制模块用来接收所有的输入数据，变换控制模块用来实现输入到输出的变换，输出控制模块用来产生所有的输出数据。

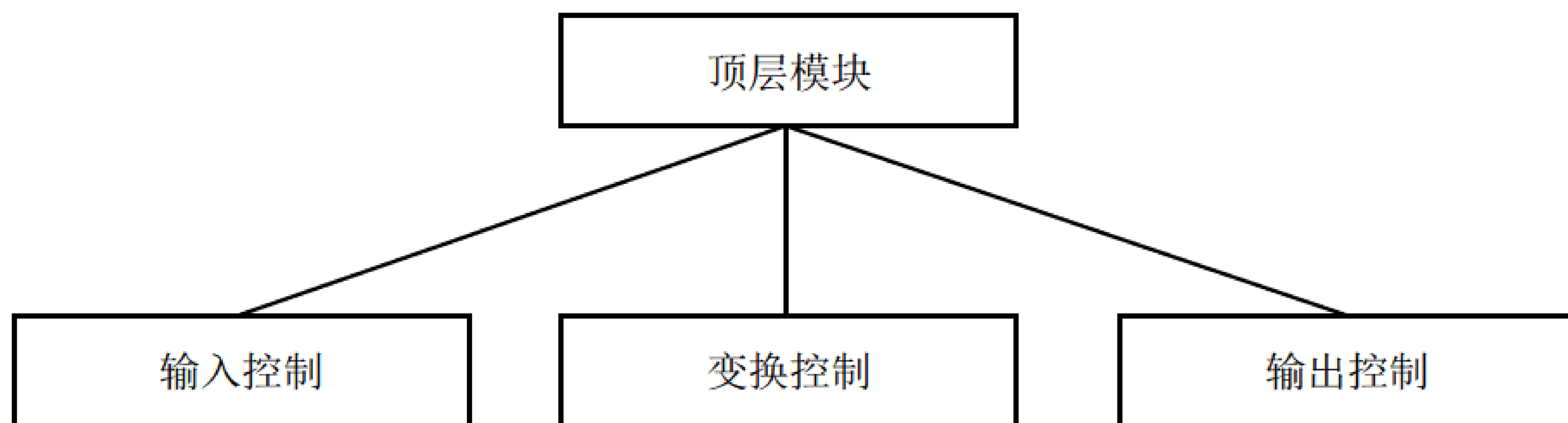


图 6-10　变换分析的第一级分解

3) 第二级分解

第二级分解主要是设计中、下层模块。

(1) 输入控制模块的分解。从变换中心的边界开始，沿着每条输入通路，把输入通路

上的每个加工映射成输入控制模块的一个低层模块。

(2) 输出控制模块的分解。从变换中心的边界开始，沿着每条输出通路，把输出通路上的每个加工映射成输出控制模块的一个低层模块。

(3) 变换控制模块的分解。变换控制模块通常没有通用的分解方法，应根据 DFD 中变换部分的实际情况进行设计。

4) 事务分析

事务分析是从事务流型 DFD 导出程序结构图。

(1) 确定事务中心和每条活动流的流特性。图 6-11 给出了事务流型 DFD 的一般形式。其中，事务中心(图中的 T)位于数条活动流的起点，这些活动流从该点呈辐射状流出。每条活动流也是一条信息流，它可以是变换流，也可以是另一条事务流。一个事务流型的 DFD 由输入流、事务中心和若干条活动流组成。

(2) 将事务流型 DFD 映射成高层的程序结构。事务流型 DFD 的高层结构如图 6-12 所示。顶层模块的功能就是整个系统的功能。接收模块用来接收输入数据，它对应于输入流。发送模块是一个调度模块，控制下层的所有活动模块。每个活动流模块对应于一条活动流，它也是该活动流映射成的程序结构图中的顶层模块。

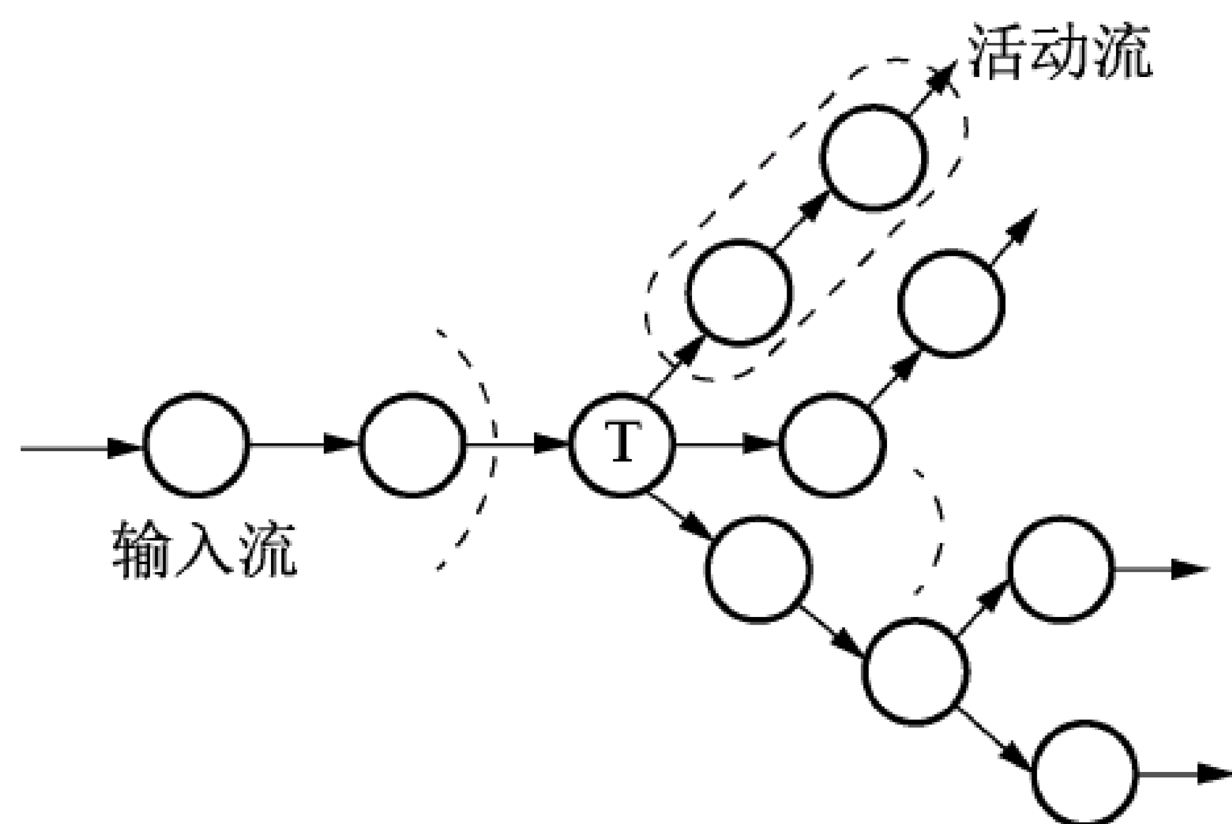


图 6-11　事务流图

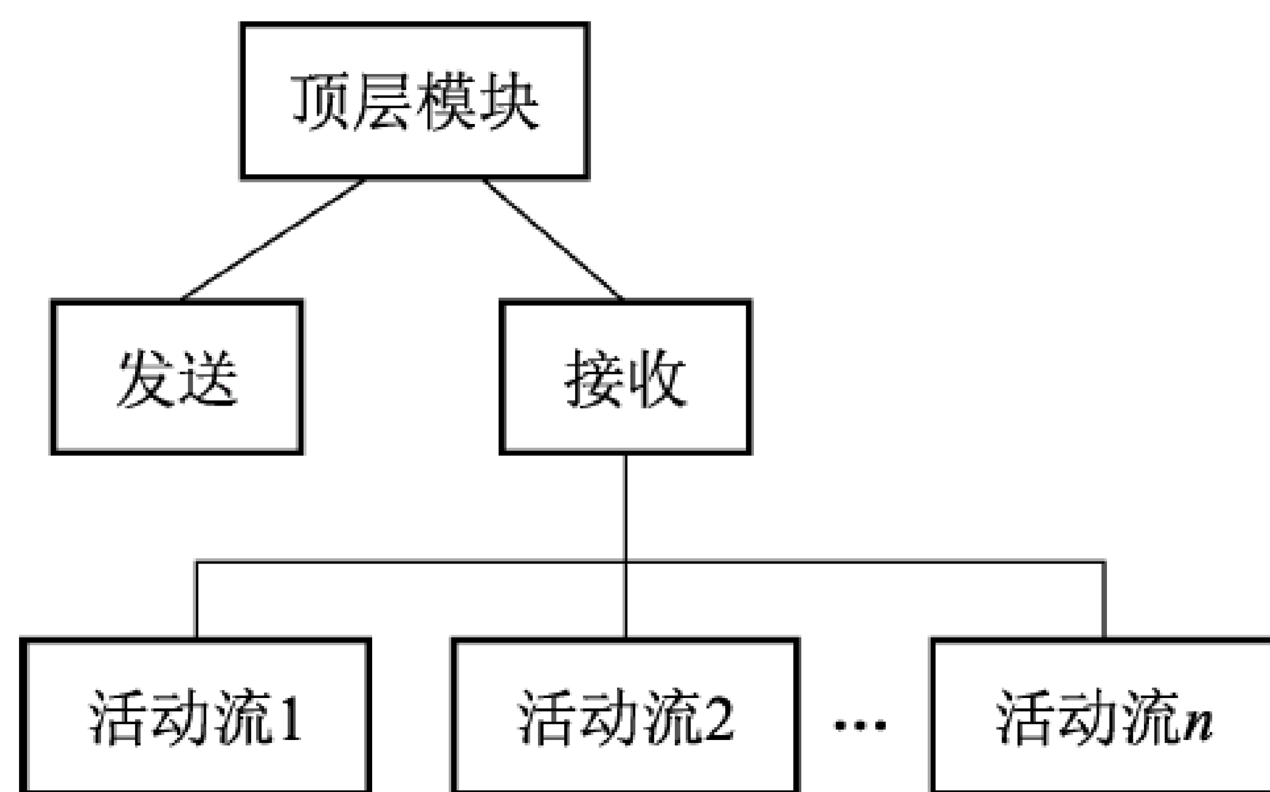


图 6-12　事务流型 DFD 的高层程序结构

(3) 进一步分解。接收模块的分解类同于变换分析中输入控制模块的分解。每个活动流模块根据其流特性(变换流或事务流)进一步采用变换分析或事务分析进行分解。

5) SD 方法的设计步骤

(1) 复查并精化数据流图。

(2) 确定 DFD 的信息流类型(变换流或事务流)。

(3) 根据流类型分别实施变换分析或事务分析。

(4) 根据系统设计的原则对程序结构图进行优化。

6.3.2　典型例题分析

例 1　以下关于结构化开发方法的叙述中，不正确的是 (15) 。(2014 年下半年试题 15)

(15) A．总的指导思想是自顶向下，逐层分解

B．基本原则是功能的分解与抽象

C．与面向对象开发方法相比，更适合大规模、特别复杂的项目

D．特别适合于数据处理领域的项目

解析：结构化系统开发方法(Structured System Development Methodology)是目前应用得最普遍的一种开发方法。其基本思想是用系统的思想和系统工程的方法，按照用户至上的原则结构化、模块化，自顶向下对系统进行分析与设计。它不限于数据处理领域项目。

答案：C

例 2　模块 A、B 和 C 包含相同的 5 个语句，这些语句之间没有联系，为了避免重复，把这 5 个模块抽取出来组成模块 D，则模块 D 的内聚类型为 (16) 内聚。(2014 年下半年试题 16)

(16) A．功能　　B．通信　　C．逻辑　　D．巧合

解析：功能内聚：完成一个单一功能，各个部分协同工作，缺一不可。

顺序内聚：处理元素相同，而且必须顺序执行。

通信内聚：所有处理元素集中在一个数据结构的区域上。

过程内聚：处理元素相关，而且必须按待定的次序执行。

瞬时内聚：所包含的任务必须在同一时间间隔内执行(如初始化模块)。

逻辑内聚：完成逻辑上相关的一组任务。

偶然内聚：完成一组没有关系或松散关系的任务。

答案：D

6.3.3　同步练习

1.　在绘制数据流图时，应遵循父图与子图平衡的原则，平衡是指______。

A．父图和子图都不得改变数据流的性质

B．子图不改变父图数据流的一致性

C．父图的输入/输出数据流与子图的输入/输出数据流一致

D．子图的输出数据流完全由父图的输入数据流确定

2.　在面向数据流的设计方法中，一般把数据流图中的数据流划分为______两种。

A．数据流和事务流　　B．变换流和数据流

C．变换流和事务流　　D．控制流和事务流

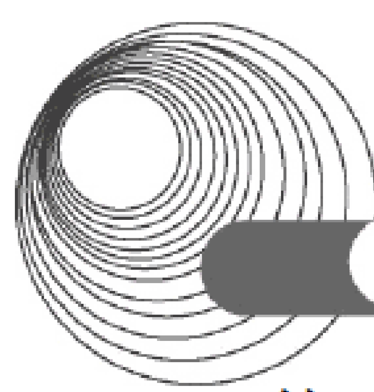
6.3.4　同步练习参考答案

1．C　　2．C

6.4　WebApp 分析与设计

6.4.1　考点辅导

WWW 的早期(大约从 1990 年到 1995 年)，Web 站点仅包含链接在一起的一些超文本文



件，这些文件使用文本和有限的图形来表示信息。随着时间的推移，一些开发工具(例如XML、Java)扩展了 HTML 的能力，使得 Web 工程师在向客户提供信息的同时也能提供计算能力。因此，基于 Web 的系统与应用(总称为 WebApp)诞生了。今天，WebApp 已经发展成为成熟的计算工具，这些工具不仅可以为最终用户提供独立的功能，而且已经同公司数据库和业务应用集成在一起了。WebApp 的特性使得大多数 WebApp 适合采用敏捷开发过程模型进行开发。

6.4.1.1 WebApp 的特性

绝大多数 WebApp 具备下列属性。

(1) 网络密集性。WebApp 驻留在网络上，服务于不同客户全体的需求。网络提供开放的访问和通信(如 Internet)或者受限的访问和通信(如企业内联网)。

(2) 并发性。大量用户可能同时访问 WebApp。很多情况下最终用户的使用模式存在很大的差异。

(3) 无法预知的负载量。WebApp 的用户数量每天都可能有数量级的变化。例如，周一显示有 100 个用户使用系统，周四就有可能会有 10000 个用户。

(4) 性能。如果一位 WebApp 用户必须等待很长时间(访问、服务器端处理、客户端格式化显示)，该用户就可能转向其他地方。

(5) 可用性。尽管期望百分之百的可用性是不切实际的，但是对于热门的 WebApp，用户通常要求能够 24/7/365(全天候)访问。

(6) 数据驱动。许多 WebApp 的主要功能是使用超媒体向最终用户提供文本、图片、音频及视频内容。除此之外，WebApp 还常被用来访问那些存储在 Web 应用环境之外的数据库中的信息。

6.4.1.2 WebApp 需求模型

WebApp 分析与传统的软件需求分析有所不同。虽然传统的软件需求分析的概念和原则全部可以应用于 Web 工程的分析活动，但 WebApp 分析的内容及参考的元素还是呈现出了其特殊性。并针对其特点，出现如面向数据的基于 ER 发展而来的 WebML、模型对象基于 UML 的 UWE 等建模方法。严格意义上讲，这些方法并非均为结构化方法，但是为了集中描述 WebApp 的建模，将其归在本章介绍。

建模以需求工程中确定的用户类别、可用目标、使用场景、业务环节等各类需求等为输入，产生如下 5 种主要的模型类型。

(1) 内容模型。

(2) 交互模型。

(3) 功能模型。

(4) 导航模型。

(5) 配置模型。

6.4.1.3 WebApp 设计

好的 WebApp 应该具有的最相关的通用特性是可用性、功能性、可靠性、效率、可维护性、安全性、可扩展性以及及时性。WebApp 的设计目标是简单性、一致性、符合性、健

壮性、导航性、视觉吸引力与兼容性。WebApp 设计根据其类型不同，适合采用混合的各种技术，进行一系列设计动作，包括：架构设计、构件设计、内容设计、导航设计、美学设计、界面设计。很多情况下，多类设计并行进行。

1. 架构设计

WebApp 描述了使 WebApp 达到其业务目标的基础结构，典型使用多层架构来构造，包括用户界面或展示层、基于一组业务规则来指导与客户端浏览器进行信息交互的控制器，以及可以包含 WebApp 的业务规则的内容层或模型层，描述将以什么方式来管理用户交互、操作内部处理任务、实现导航及展示内容。模型-视图-控制器(Model-View-Controller，MVC)结构是 WebApp 基础结构模型之一，它将 WebApp 功能及信息内容分离。

2. 构件设计

在 WebApp 中，内容和功能的界限通常并不清晰，因此首先明确 WebApp 构件：①定义良好的聚合功能，为最终用户处理内容或提供计算或处理数据；②内容和功能的聚合包，提供最终用户所需要的功能。因此，WebApp 构件设计通常包括内容设计元素和功能设计元素。

(1) 构件级内容设计。关注内容对象，以及包装后展示给最终用户的方式，应该适合创建的 WebApp 特性。一般情况下，内容对象不需要被组织成构件，加以分别实现。但是，随着 WebApp、内容对象及其关系的规模和复杂度的增长，组织内容是必要的。

(2) 构件级功能设计。将 WebApp 作为一系列构件加以交付，这些构件与信息体系结构并行开发，以确保一致性。并且需要在一开始就考虑需求模型和初始信息架构，然后再进一步考查功能如何影响用户与系统的交互、要展示的信息以及要管理的用户任务。在架构设计中，往往将 WebApp 的内容和功能结合在一起来设计功能架构，描述关键的功能构件及其交互。

3. 内容设计

WebApp 的内容结构(线性或非线性)也影响架构，因此设计内容体系结构。内容体系结构着重于内容对象(诸如网页的组成对象)的表现和导航的组织，通常采用线性结构、网格结构、层次结构、网络结构四种结构及其组合。当内部交互可预测顺序很常见时，采用线性结构，如帮助文档、产品订单输入顺序等。当 WebApp 内容按类别组织成十分规则的二维或多维时，可以采用网格结构，如销售网站水平代表要出售的商品准类，垂直表示制造商。层次结构可以设计成是控制流水平地穿过垂直分支(通过超文本)的方式，即同层之间可以通过超链接相连。网络结构一般是对网页进行设计，使其能够将控制通过超链接传递到系统中几乎所有的网页。将上述设计结构进行组合，就形成复合结构。

4. 导航设计

建立了 WebApp 的体系结构及其构件后，定义导航路径，使用户可以访问 WebApp 的内容和功能。为每一类用户角色定义一组类，包含一个或多个内容对象或者包含 WebApp 的功能。当用户与 WebApp 进行交互时，会接触到一系列导航语义单元，即信息(导航节点)和相关的导航结构(导航链接)的集合，它们相互协作共同完成相关的用户请求的一部分。定义导航机制，如导航链接，水平或垂直导航条(列表)，标签或者一个完整的站点地图入口。

用户界面是围绕着运行在客户端上的浏览器的特性进行设计的，数据层位于服务器上。

6.4.2 典型例题分析

例 面向对象分析的第一步是__(37)__。(2011 年上半年试题 37)

(37) A．定义服务　　B．确定附加的系统约束
　　C．确定问题域　　D．定义类和对象

解析：面向对象分析(OOA)需要将真实世界进行抽象，通过问题的叙述，将真实世界系统加以描述。分析的目的是为了构造一个系统属性和系统行为的模型，该模型是根据对象和对象之间的关系、动态控制和功能转移来确定的。OOA 应该包含以下几个步骤。

① 分析问题域，建立用例模型。

② 发现和定义对象和类。

③ 识别对象的内部特征。

④ 识别对象的外部联系。

⑤ 识别对象之间的交互。

面向对象分析的基础就是问题域以及用户的需求，研究问题域和系统需求的主要目的是通过对问题域的深入研究，建立一个能够满足用户需求的系统模型。面向对象分析的第一步就是分析用户需求，确定问题域，在此基础上建立用例模型。

答案：C

6.4.3 同步练习

面向对象分析与设计中的__(1)__是指一个模块在可扩展性方面应该是开放的，而在更改性方面应该是封闭的；而__(2)__是指子类应当可以替换父类并出现在父类能够出现的任何地方。

(1)～(2) A．开放封闭原则　　B．替换原则
　　C．依赖原则　　D．单一职责原则

6.4.4 同步练习参考答案

(1) A　　(2) C

6.5 用户界面设计

6.5.1 考点辅导

用户界面(UI)设计在人与计算机之间搭建了一个有效的交流媒介。遵循一系列的界面设

计原则定义界面对象和界面设计动作，然后创建构成用户界面原型基础的屏幕布局。

6.5.1.1 用户界面设计的黄金原则

Theo Mandel 在其关于界面设计的著作中提出了 3 条“黄金原则”：用户操纵控制；减少用户的记忆负担；保持界面一致。

这些黄金原则实际上构成了一系列用户界面设计原则的基础，这些原则可以指导软件设计的重要方面。

1. 用户操纵控制

在很多情况下，设计者为了简化界面的实现可能会引入约束和限制，其结果可能是界面易于构建，但会妨碍使用。Mandel 定义了一组设计原则，允许用户操纵控制。

1) 以不强迫用户进入不必要的或不希望的动作的方式来定义交互模式

交互模式就是界面的当前状态。例如，如果在字处理软件的菜单中选择拼写检查，则软件将转移到拼写检查模式。如果用户希望在这种情形下进行一些文本编辑，则没有理由强迫用户停留在拼写检查模式，用户应该能够几乎不需要做任何动作就能进入和退出该模式。

2) 提供灵活的交互

由于不同的用户有不同的交互偏好，因此应该提供选择机会。例如，软件可能允许用户通过键盘命令、鼠标移动、数字笔、多触摸屏或语音识别命令等方式进行交互。但是，每个动作并非要受控于每一种交互机制。例如，考虑使用键盘命令(或语音输入)来画一幅复杂形状的图形是有一定难度的。

3) 允许中断和撤销用户交互

即使当陷入到一系列动作之中时，用户也应该能够中断动作序列去做某些其他事情(而不会失去已经做过的工作)，用户也应该能够“撤销”任何动作。

4) 当技能级别增长时可以使交互流线化并允许定制交互

用户经常发现他们重复地完成相同的交互序列。因此，值得设计一种“宏”机制，使得高级用户能够定制界面以方便交互。

5) 使用户与内部技术细节隔离开来

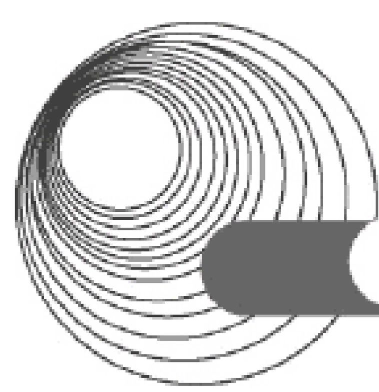
用户界面应该能够将用户移入到应用的虚拟世界中来。用户不应该知道操作系统、文件管理功能或其他神秘的计算技术。其实，界面不应该要求用户在机器内部层次上进行交互(例如，不应该要求用户在应用软件中输入操作系统命令)。

6) 设计应允许用户与出现在屏幕上的对象直接交互

当用户能够操纵完成某任务所必需的对象，并且以一种该对象好像是真实物理存在的方式来操作它时，用户就会有一种控制感。例如，某应用界面可允许用户“拉伸”某对象(增大其尺寸)，就是直接操纵的一种实现。

2. 减轻用户的记忆负担

用户必须记住的东西越多，与系统交互时，出错的可能性也就越大。因此，一个经过精心设计的用户界面不会加重用户的记忆负担。只要有可能，系统应该“记住”有关的信息，并通过能够帮助回忆的交互场景来帮助用户。Mandel 定义了一组设计原则，使得界面能够减少用户的记忆负担。



1) 减少对短期记忆的要求

当用户陷于复杂的任务时，短期记忆的要求将会很大。界面的设计应该尽量不要求记住过去的动作、输入和结果。可行的解决办法是通过提供可视的提示，使得用户能够识别过去的动作，而不是必须记住它们。

2) 建立有意义的默认

初始的默认集合应该对一般的用户有意义，但是，用户应该能够说明个人的偏好。然而，“reset(重置)”选择应该是可用的，使得可以重新定义初始默认值。

3) 定义直观的快捷方法

当使用助记符来完成系统功能时，助记符应该以容易记忆的方式联系到相关动作。

4) 界面的视觉布局应该基于真实世界的象征

例如，一个账单支付系统应该使用支票簿和支票登记簿来指导用户的账单支付过程。这使得用户能够依赖于很好理解的可视提示，而不是记住复杂难懂的交互序列。

5) 以不断进展的方式揭示信息

界面应该以层次化的方式进行组织，即关于某任务、对象或某行为的信息应该首先在高抽象层次上呈现。更多的细节中十分常见的一个功能是加下划线，该功能本身是“文本风格”菜单下多个功能中的一个。然而，每种加下划线的功能并未列出，用户必须选择加下划线，然后所有加下划线的选项(如加单下划线、加双下划线、加虚下划线)才被展示出来。

3. 保持界面一致

用户应该以一致的方式展示和获取信息，这意味着：按照贯穿所有屏幕显示的设计规则来组织可视信息；将输入机制约束到有限的集合，在整个应用系统中得到一致的使用；从任务到任务的导航机制要一致地定义和实现。Mandel 定义了一组帮助保持界面一致性的设计原则。

1) 允许用户将当前任务放入有意义的环境中

很多界面使用数十个屏幕图像来实现复杂的交互层次，提供指示器(例如，窗口标题、图标、一致的颜色编码)帮助用户知道当前工作环境是十分重要的。另外，用户应该能够确定它来自何处以及存在什么途径转换到新任务。

2) 在应用系统家族内保持一致性

一组应用系统(或一套产品)应使用相同的设计规则，以保持所有交互的一致性。

3) 如果过去的交互模型已经建立起了用户期望，除非有不得已的理由，否则不要改变它一个特殊的交互序列一旦已经变成事实上的标准，则用户在遇到的每个应用系统中均会如此期望，如果改变，将导致混乱。

6.5.1.2 用户界面设计的分析与设计

用户界面的分析与设计过程始于创建不同的系统功能模型(从外部看对系统的感觉)，用于完成系统功能的任务被分为面向人的和面向计算机的，考虑那些应用到界面设计中的各种设计问题，各种工具被用于建造原型并最终实现设计模型，最后由最终用户从质量的角度对结果进行评估。

1. 用户界面分析和设计模型

(1) 软件工程师所创建的设计模型(Design Model)。整个系统设计模型包括对软件的数据结构、体系结构、界面和过程的表示。界面设计往往是设计模型的附带结果。

(2) 人机界面设计工程师创建的用户模型(User Model)。用户模型描述系统最终用户的特点。在设计前，应首先对用户进行分类，了解用户的特点，包括年龄、性别、实际能力、教育、文化和种族背景、动机、目的以及个性等。

(3) 最终用户在脑海里对界面产生的映像，称为用户的心理模型或系统感觉(SystemPerception)。系统感觉是最终用户主观想象的系统映像，描述了期望的系统能提供的操作，其描述的精确程度依赖于最终用户对软件的熟悉程度。

(4) 系统实现者创建的系统映像(System Image)。系统映像包括基于计算机的系统的外在表示(界面的观感)和用来描述系统语法和意义的支撑信息(书、手册、录像带、帮助文件等)。如果系统映像和系统感觉是一致的，用户就会感觉软件很舒服，使用起来很有效。为了融合这些模型，设计模型必须适应包含在用户模型中的信息，并且系统映像必须准确地反映接口的语法和语义。

2. 用户界面分析和设计的过程

用户界面的分析与设计过程是迭代的，包括 4 个不同的框架活动：界面分析及建模、界面设计、界面构造和界面确认。

界面分析活动的重点在于那些与系统交互的用户的轮廓，记录技能级别、业务理解以及对新系统的一般感悟，并定义不同的用户类型，对每个用户类别进行需求引导。

界面设计的目标是定义一组界面对象和动作(以及它们的屏幕表示)，使得用户能够以满足系统所定义的每个使用目标的方式完成所有定义的任务。

界面构造通常开始于创建可评估使用场景的原型。随着迭代设计过程的继续，用户界面开发工具可以用来完成界面的构造。

界面确认着重于以下几点：界面正确地实现每个用户任务的能力、适应所有任务变化的能力以及达到所有一般用户需求的能力；界面容易使用和学习的程度；用户将界面作为其工作中有用工具的接受程度。

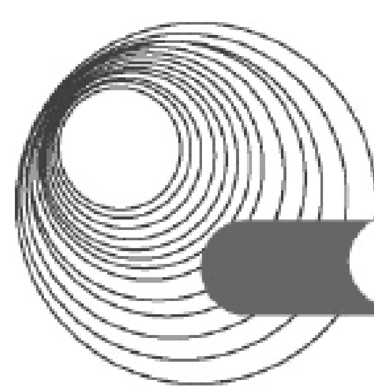
6.5.1.3　用户界面设计问题

在进行用户界面设计时，几乎总会遇到以下 4 个问题：系统响应时间、帮助设施、错误信息处理、菜单和命令标记。

1. 系统响应时间

系统响应时间不能令人满意是交互式系统用户经常抱怨的问题。一般来说，系统响应时间是指从用户开始执行动作到软件以预期的输出和动作形式给出响应这段时间。

系统响应时间包括两方面的属性：时间长度和可变性。如果系统响应时间过长，用户就会感到焦虑和沮丧。系统时间的可变性是指相对于平均时间的偏差，在很多情况下这是最重要的响应时间特性。即使响应时间比较长，响应时间的低可变性也有助于用户建立稳定的交互节奏。例如，稳定在 1s 的命令响应时间，比从 0.1s 到 2.5s 不定的响应时间要好。



2. 帮助设施

几乎所有计算机交互式系统的用户都时常需要帮助，考虑帮助设施时需要在设计中解决以下问题。

(1) 进行系统交互时，是否在任何时候对任何系统功能都能得到帮助？有两种选择：提供部分功能与动作的帮助和提供全部功能的帮助。

(2) 用户怎样请求帮助？有 3 种选择：帮助菜单、特殊功能键和 HELP 命令。

(3) 如何表达帮助？有 3 种选择：提供单独的帮助窗口、在另一个窗口中指示参考某个已印刷的文档(不是理想方式)或在屏幕特定位置给出一行或两行的简单提示。

(4) 用户如何回到正常的交互方式？可做的选择包括屏幕上显示的返回按钮、功能键或控制序列。

(5) 如何构造帮助信息？有 3 种选择：平面结构(所有信息均通过关键词来访问)、分层结构(用户可以进入分层结构得到更详细的信息)和超文本的使用。

3. 错误信息处理

出错信息和警告是指出现问题时系统反馈给用户的“坏消息”。如果做不好，出错信息和警告会给出无用和误导的信息，反而增加了用户的沮丧感。通常，交互式系统给出的出错信息和警告应具备以下特征。

(1) 消息以用户可以理解的语言描述问题。

(2) 消息应提供如何从错误中回复的建设性意见。

(3) 消息应指出错误可能导致哪些不良后果(例如破坏数据文件)，以便用户检查是否出现了这些情况。

(4) 消息伴随着视觉或听觉上的提示。也就是说，显示消息应该伴随警告声或者消息用闪烁方式显示，或以明显表示错误的颜色来显示。

(5) 消息不应是裁判性的，即不能指责用户。

4. 菜单和命令标记

输入命令曾经是用户和系统交互的主要方式，并广泛用于各种应用程序中。现在，面向窗口的界面采用点击和选取方式，减少了用户对输入命令的依赖。但许多高级用户仍然喜欢面向命令的交互方式。在提供命令或菜单标签交互方式时，必须考虑以下问题。

(1) 每个菜单选择是否都有对应的命令？

(2) 以何种方式提供命令？有 3 种选择：控制序列(如 Alt+P 快捷键)、功能键或输入命令。

(3) 学习和记忆命令的难度有多大？忘记命令怎么办？

(4) 用户是否可以定制和缩写命令？

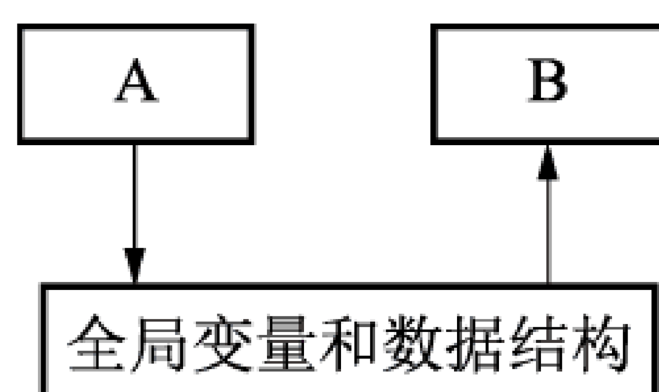
(5) 在界面环境中菜单标签是不是自解释的？

(6) 子菜单是否与主菜单所指的功能相一致？

6.5.2 典型例题分析

例 如下图所示，模块 A 和模块 B 都访问相同的全局变量和数据结构，则这两个模块

之间的耦合类型为 (29) 耦合。(2016 年上半年上午试题 29)



(29) A．公共　　B．控制　　C．标记　　D．数据

答案：A

解析：公共耦合指通过一个公共数据环境相互作用的那些模块间的耦合。公共数据环境可以是全局变量或数据结构，共享的通信，内存的公共覆盖区及任何存储介质上的文件。

6.5.3　同步练习

1．在 UML 的各种视图中， (1) 显示外部参与者观察到的系统功能； (2) 从系统的静态结构和动态行为角度显示系统内部如何实现系统的功能； (3) 显示的是源代码以及实际执行代码的组织结构。

(1)～(3) A．用例视图　　B．进程视图
C．实现视图　　D．逻辑视图

2．采用 UML 进行软件设计时，可用_____关系表示两类事物之间存在的特殊/一般关系，用聚集关系表示事物之间存在的整体/部分关系。

A．依赖　　B．聚集　　C．泛化　　D．实现

6.5.4　同步练习参考答案

1．(1) A　(2) D　(3) C

2．C

6.6　本 章 小 结

结构化方法由结构化分析、结构化设计、结构化程序设计构成，它是一种面向数据流的开发方法。结构化分析是根据分解与抽象的原则，按照系统中数据处理的流程，用数据流图来建立系统的功能模型，从而完成需求分析工作。结构化设计是根据模块独立性准则、软件结构优化准则将数据流图转换为软件的体系结构，用软件结构图来建立系统的物理模型，实现系统的概要设计。结构化程序设计使用 3 种基本控制结构构造程序，任何程序都可以由顺序、选择和重复 3 种基本控制结构构造。

结构化方法总的指导思想是自顶向下、逐层分解，它的基本原则是功能的分解与抽象。它是软件工程中最早出现的开发方法，特别适合于数据处理领域的问题，但是不适合解决大规模的、特别复杂的项目，且难以适应需求的变化。

6.7 达标训练题及参考答案

6.7.1 达标训练题

1. 以下关于数据流图中基本加工的叙述，不正确的是______。
   A. 对每一个基本加工，必须有一个加工规格说明
   B. 加工规格说明必须描述把输入数据流变换为输出数据流的加工规则
   C. 加工规格说明必须描述实现加工的具体流程
   D. 决策表可以用来表示加工规格说明

2. 在划分模块时，一个模块的作用范围应该在其控制范围之内。若发现其作用范围不在其控制范围内，则______不是适当的处理方法。
   A. 将判定所在模块合并到父模块中，使判定处于较高层次
   B. 将受判定影响的模块下移到控制范围内
   C. 将判定上移到层次较高的位置
   D. 将父模块下移，使该判定处于较高层次

3. 逆向工程从源代码或目标代码中提取设计信息，通常在原软件生命周期的__(3)__阶段进行。
   A. 需求分析　　B. 软件设计
   C. 软件实现　　D. 软件维护

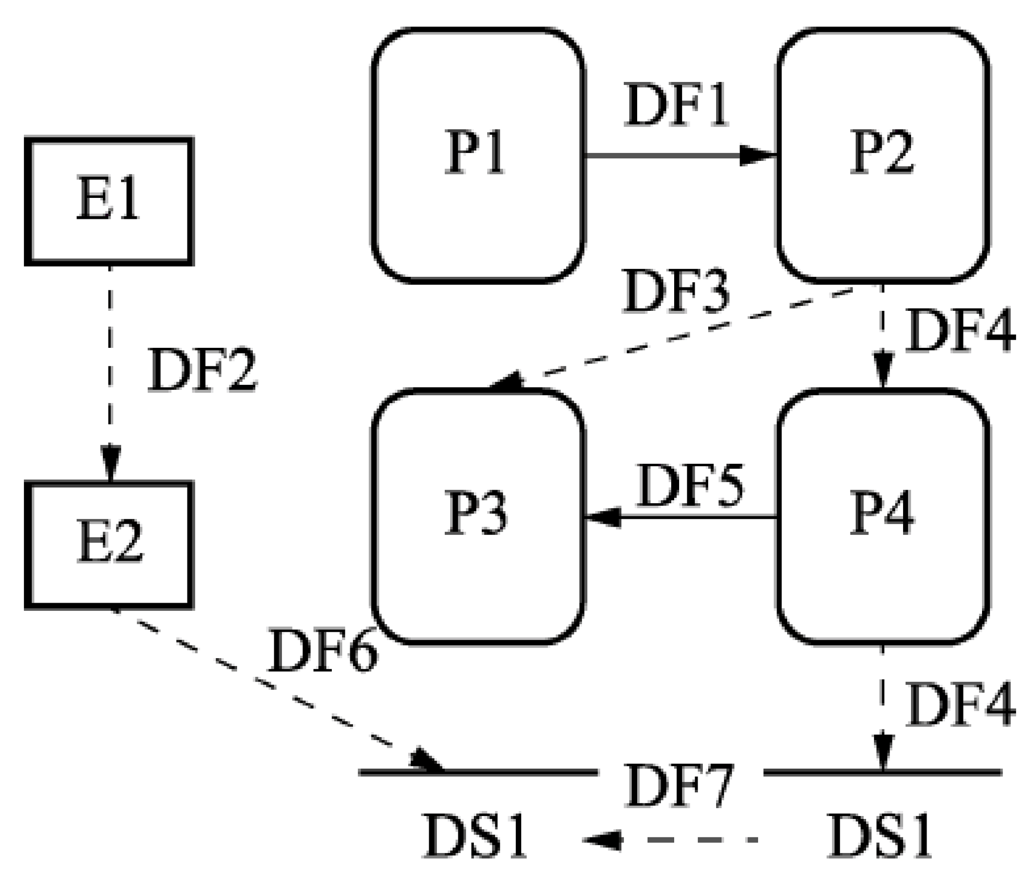
4. 一个程序根据输入的年份和月份计算该年中该月的天数，输入参数包括年份(正整数)、月份(用 1～12 表示)。若用等价类划分测试方法进行测试，则______不是一个合适的测试用例(分号后表示测试的输出)。
   A. (2013,1;31)　　B. (0,1;'错误')
   C. (0,13;'错误')　　D. (2000,-1; '错误')

5. ______不是单元测试主要检查的内容。
   A. 模块接口　　B. 局部数据结构
   C. 全局数据结构　　D. 重要的执行路径

6. “软件产品必须能够在 3 秒内对用户请求做出响应”属于软件需求中的______。
   A. 功能需求　　B. 非功能需求
   C. 设计约束　　D. 逻辑需求

7. 某项目为了修正一个错误而进行了修改。错误修改后，还需要进行______以发现这一修改是否引起原本正确运行的代码出错。
   A. 单元测试　　B. 接收测试
   C. 安装测试　　D. 回归测试

8. 在下图所示数据流图中，共存在______个错误。



(8) A．4　　　　B．6　　　　C．8　　　　D．9

9．阅读下列说明和图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

【说明】

某大学欲开发一个基于 Web 的课程注册系统，该系统的主要功能如下。

1) 验证输入信息

(1) 检查学生信息：检查学生输入的所有注册所需信息。如果信息不合法，返回学生信息不合法提示；如果合法，输出合法学生信息。

(2) 检查学位考试结果：检查学生提供的学位考试结果。如果不合法，返回学位考试结果不合法提示；如果合法，检查该学生注册资格。

(3) 检查学生注册资格：根据合法学生信息和合法学位考试结果，检查该学生对欲选课程的注册资格。如果无资格，返回无注册资格提示；如果有注册资格，则输出注册学生信息(包含选课学生标识)和欲注册课程信息。

2) 处理注册申请

(1) 存储注册信息：将注册学生信息记录在学生库。

(2) 存储所注册课程：将选课学生标识与欲注册课程信息进行关联，然后存入课程库。

(3) 发送注册通知：从学生库中读取注册学生信息，从课程库中读取所注册课程信息，给学生发送接受提示；给教务人员发送所注册课程信息和已注册学生信息。

现采用结构化方法对课程注册系统进行分析与设计，获得如图 6-13 所示的 0 层数据流图和图 6-14 所示的 1 层数据流图。

【问题 1】(2 分)

使用说明中的词语，给出图 6-13 中的实体 E1 和 E2 的名称。

【问题 2】(2 分)

使用说明中的词语，给出图 6-14 中的数据存储 D1 和 D2 的名称。

【问题 3】(8 分)

根据说明和图中术语。补充图 6-14 中缺失的数据流及其起点和终点。

【问题 4】(3 分)

根据补充完整的图 6-13 和图 6-14，说明上层的哪些数据流是由下层的哪些数据流组合而成。

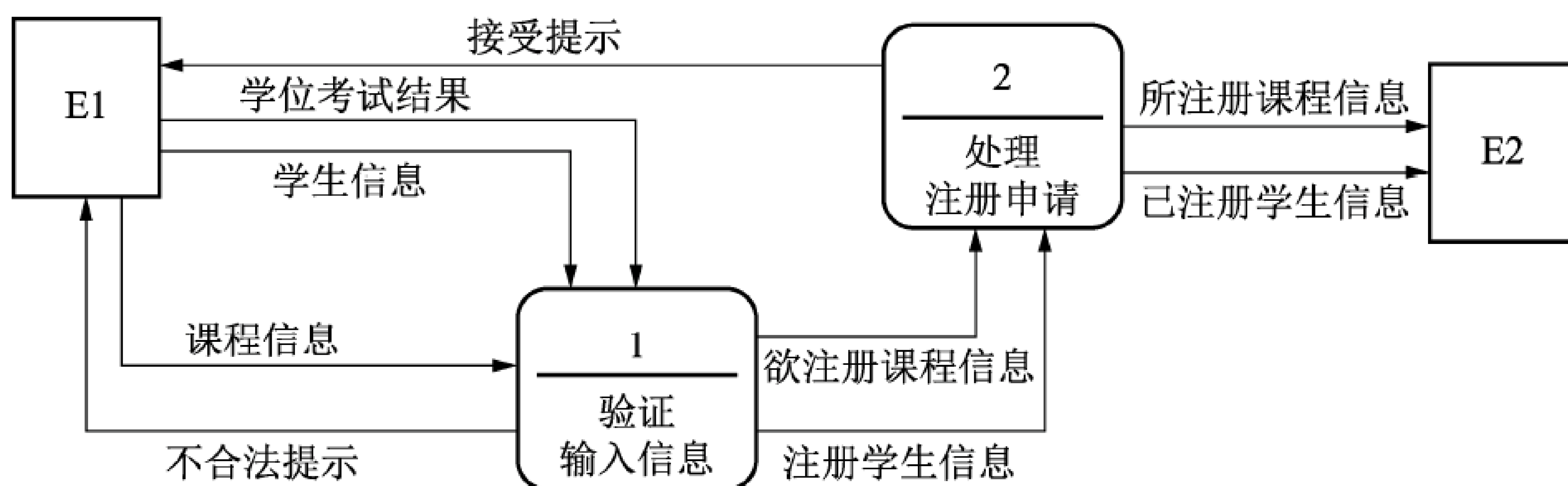


图 6-13　0 层数据流图

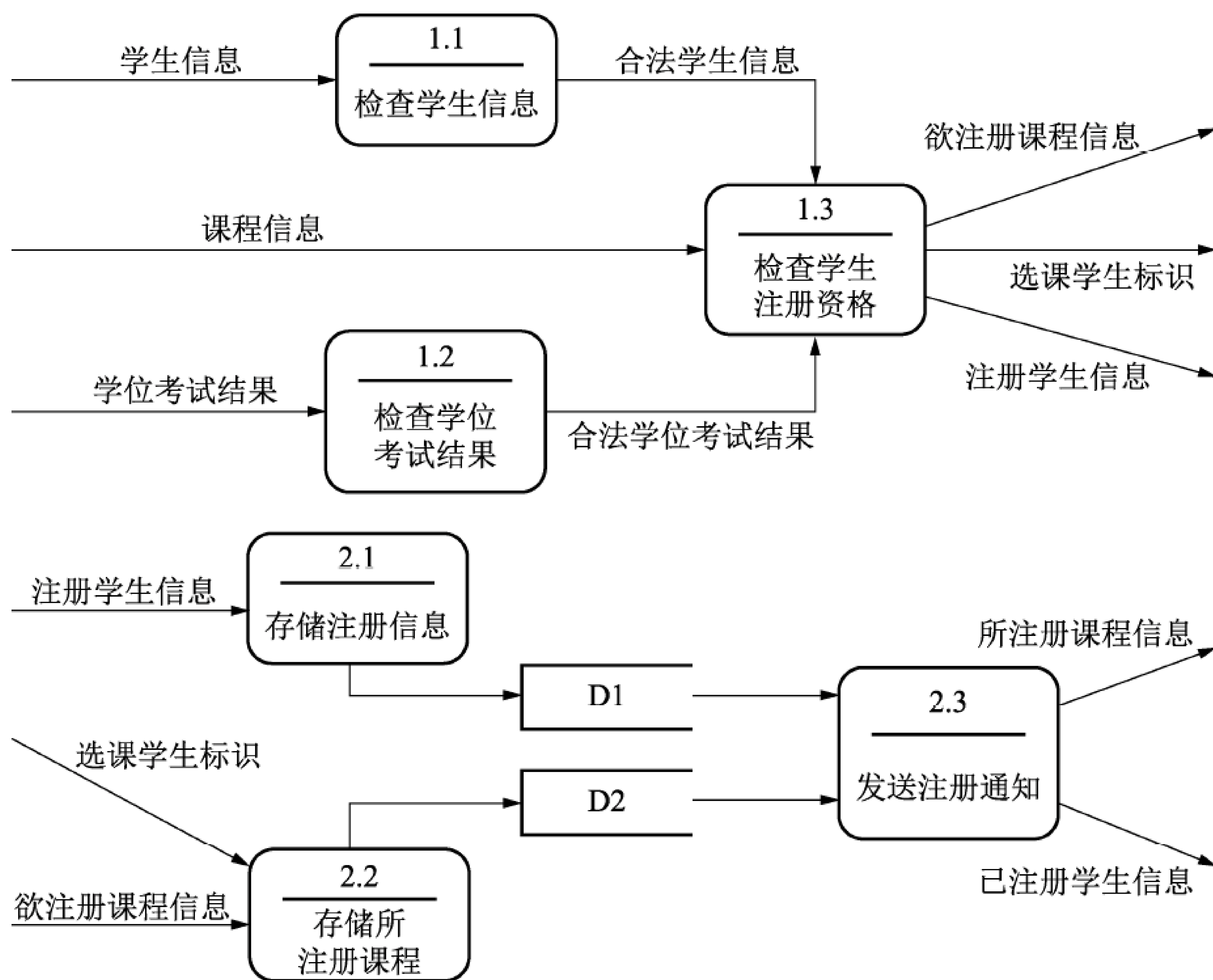


图 6-14　1 层数据流图

10. 阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某慈善机构欲开发一个募捐系统，以跟踪记录为事业或项目向目标群体进行募捐而组织的集体性活动。该系统的主要功能如下所述。

(1) 管理志愿者。根据募捐任务给志愿者发送加入邀请、邀请跟进、工作任务；管理志愿者提供的邀请响应、志愿者信息、工作时长、工作结果等。

(2) 确定募捐需求和收集所募捐赠(资金及物品)。根据需求提出募捐任务、活动请求和捐赠请求，获取所募集的资金和物品。

(3) 组织募捐活动。根据活动请求，确定活动时间范围。根据活动时间，搜索场馆，即：向场馆发送场馆可用性请求，获得场馆可用性。然后根据活动时间和地点推广募捐活动，根据相应的活动信息举办活动，从募捐机构获取资金并向其发放赠品。获取和处理捐

赠，根据捐赠请求，提供所募集的捐赠；处理与捐赠人之间的交互，即：录入捐赠人信息，处理后存入捐赠人信息表；从捐赠人信息表中查询捐赠人信息，向捐赠人发送募捐请求，并将已联系的捐赠人存入已联系的捐赠人表。根据捐赠请求进行募集，募得捐赠后，将捐赠记录存入捐赠表；对捐赠记录进行处理后，存入已处理捐赠表，向捐赠人发送致谢函，根据已联系的捐赠人和捐赠记录进行跟踪，将捐赠跟进情况发送给捐赠人。

先采用结构化方法对募捐系统进行分析与设计，获得如图 6-15、图 6-16 所示分层数据流图。

【问题 1】(4 分)

使用说明中的词语，给出图 6-15 中的实体 E1～E4 的名称。

【问题 2】(7 分)

在建模 DFD 时，需要对有些复杂加工(处理)进行进一步精化，图 6-16 为图 6-15 中处理 3 的进一步细化的 1 层数据流图，图 6-17 为图 6-16 中 3.1 进一步细化的 2 层数据流图。补全图 6-16 中加工 P1、P2 和 P3 的名称和图 6-16 与图 6-17 中缺少的数据流。

【问题 3】(4 分)

使用说明中的词语，给出图 6-17 中的数据存储 D1～D4 的名称。

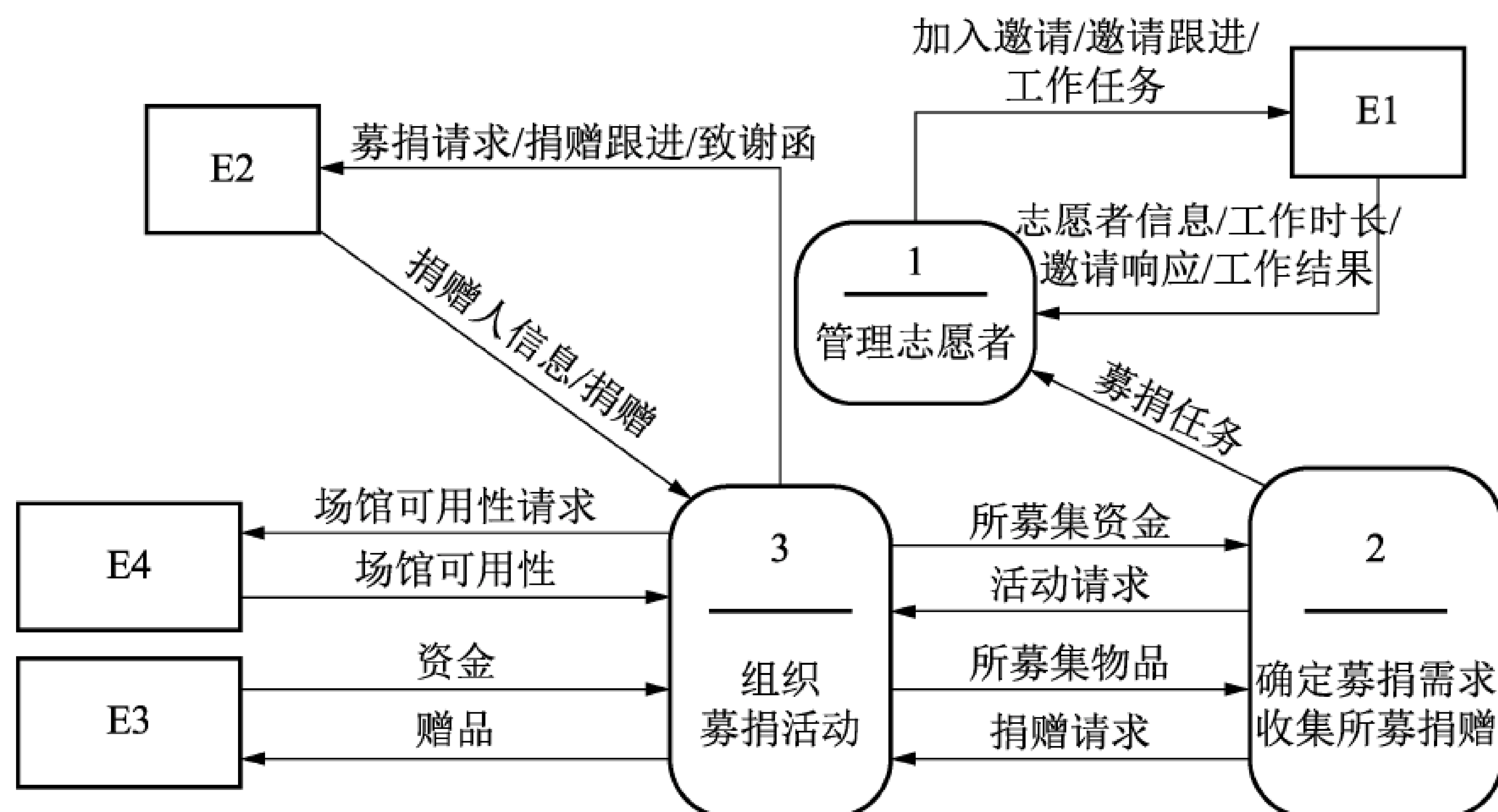


图 6-15　0 层数据流图

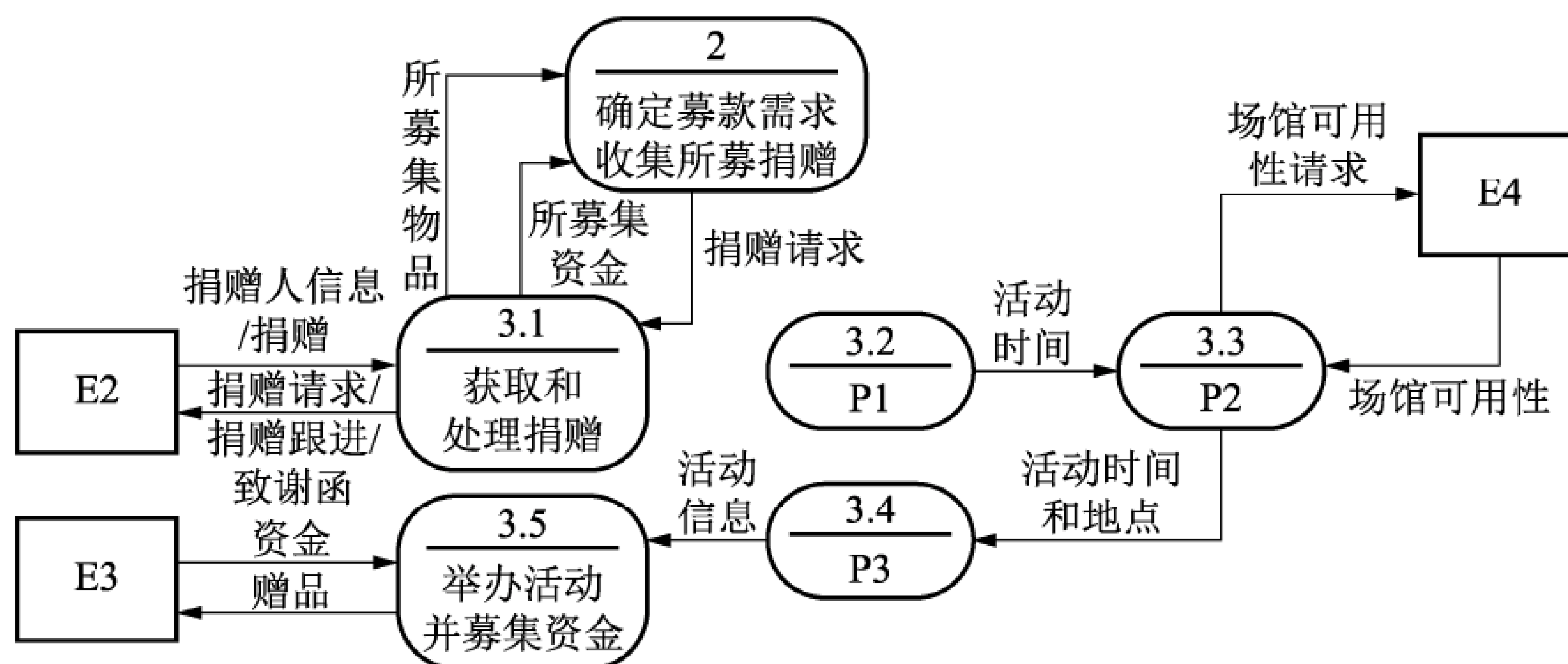


图 6-16　1 层数据流图

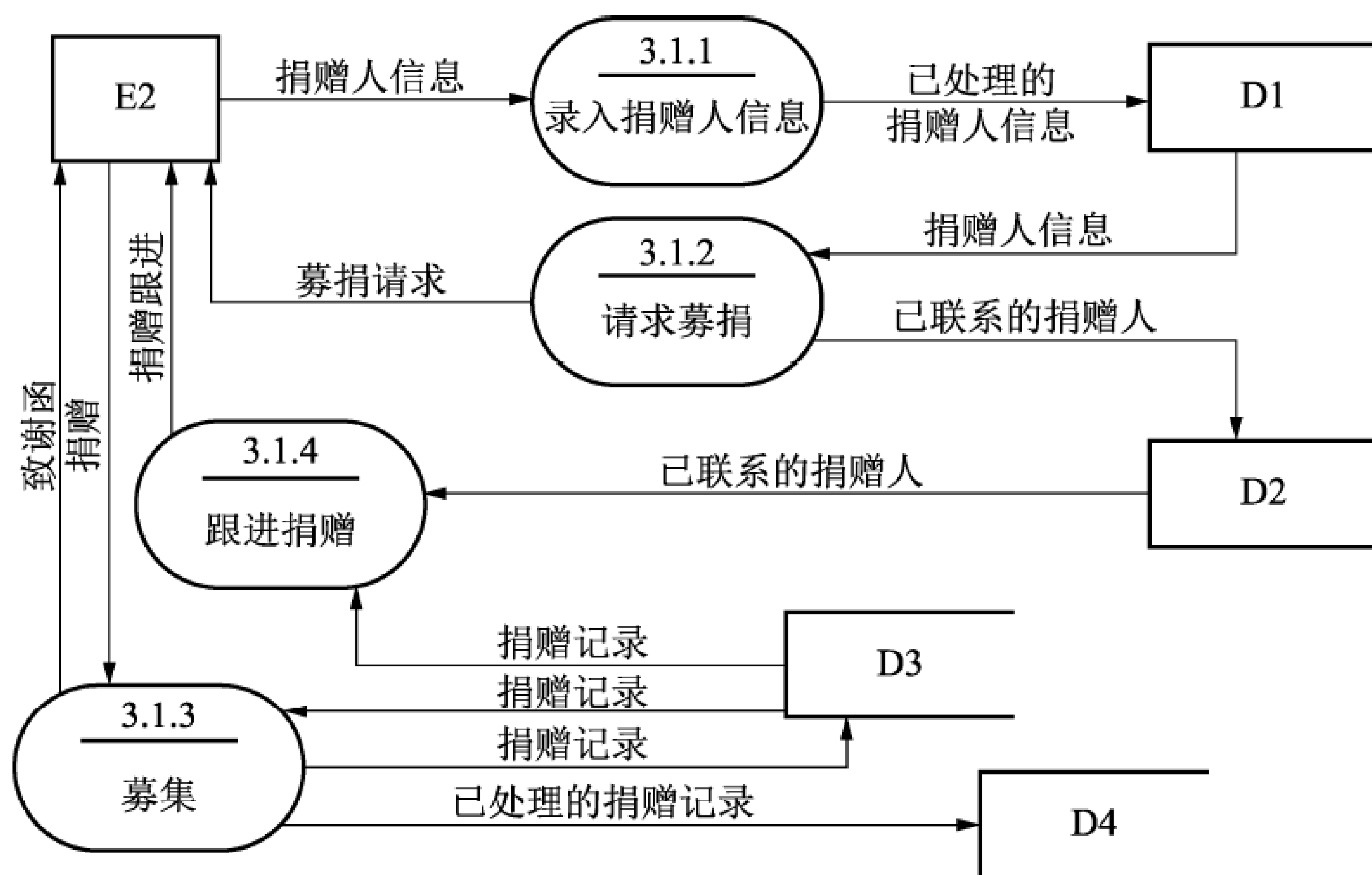


图 6-17　2 层数据流图

6.7.2　参考答案

1．C　　2．D　　3．D　　4．C

5．C　　6．B　　7．D　　8．B

9．【问题 1】

E1：学生　　E2：教务人员

【问题 2】

D1：学生库　　D2：课程库

【问题 3】

缺失的数据流：学生信息不合法提示；起点：检查学生信息；终点：E1 或学生

缺失的数据流：学位考试结果不合法提示；起点：检查学位考试结果；终点：E1 或学生

缺失的数据流：无注册资格提示；起点：检查学生注册资格；终点：E1 或学生

缺失的数据流：接受提示；起点：发送注册通知；终点：E1 或学生

【问题 4】

0 层数据流图中的“不合法提示”数据流是由 1 层数据流图中的“学生信息不合法提示”“学位考试结果不合法提示”和“无注册资格提示”组合而成。

10．【问题 1】

E1：志愿者　E2：捐赠人　E3：募捐机构　E4：场馆

【问题 2】

P1：确定活动时间范围　P2：搜索场馆　P3：推广募捐活动

图 6-16 中缺少的数据流如下。名称：活动请求；起点：确定募款需求、收集所募捐赠；终点：P1

图 6-17 中缺少的数据流如下。名称：捐赠请求；起点：E2；终点：募集

【问题 3】

D1：捐赠人信息表　D2：已联系的捐赠人表　D3：捐赠表　D4：已处理捐赠表

第 7 章　面向对象技术

大纲要求：

- 面向对象开发概念(类、对象、属性、封装性、继承性、多态性、对象之间的引用)。
- 面向对象开发方法的优越性以及有效领域。
- 面向对象设计方法(体系结构、类的设计、用户接口设计)。
- 面向对象实现方法(选择程序设计语言、类的实现、方法的实现、用户接口的实现、准备测试数据)。
- 面向对象程序设计语言(如 C++、Java、Visual C++)的基本机制。
- 面向对象数据库、分布式对象的概念。

7.1　面向对象的基本概念

7.1.1　考点辅导

Peter Coad 和 Edward Yourdon 提出用下面的等式识别面向对象方法，即

面向对象＝对象+分类+继承+通过消息的通信

可以说，采用这 4 个概念开发的软件系统是面向对象的。

1. 对象

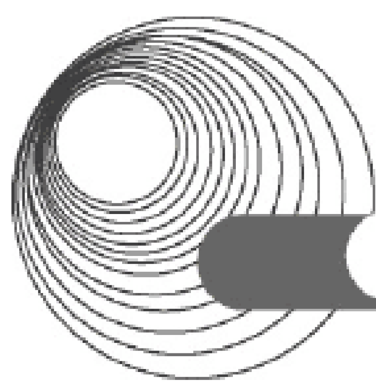
在面向对象的系统中，对象是基本的运行时的实体，它既包括数据(属性)，也包括作用于数据的操作(行为)，所以一个对象把属性和行为封装为一个整体。封装是一种信息隐蔽技术，它的目的是使对象的使用者和生产者分离，使对象的定义和实现分开。从程序设计者来看，对象是一个程序模块；从用户来看，对象为他们提供了所希望的行为。在对象内的操作通常叫做方法。一个对象通常可由对象名、属性和操作三部分组成。

2. 消息

对象之间进行通信的一种构造叫做消息。但一个消息发送给某个对象时，包含要求接收对象去执行某些活动的信息，接收到消息的对象经过解释，然后予以响应，这种通信机制叫做消息传递。发送消息的对象不需要知道接收消息的对象如何对请求予以响应。

3. 类

一个类定义了一组大体上相似的对象，一个类所包含的方法和数据描述一组对象的共同行为和属性。类是在对象之上的抽象，对象是类的具体化、是类的实例。通常把一个类和这个类的所有对象称为“类及对象”或对象类。



4. 继承

继承是父类和子类之间共享数据和方法的机制。这是类之间的一种关系，在定义和实现一个类的时候，可以在一个已经存在的类的基础上进行，把这个已经存在的类所定义的内容作为自己的内容，并加入若干新的内容。

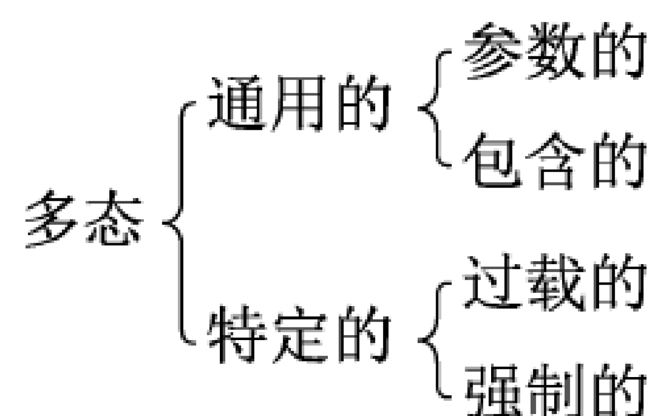
一个父类可以有多个子类，这些子类都是父类的特例，父类描述了这些子类的公共属性和操作。一个子类可以继承它的父类(或祖先类)中的属性和操作，这些属性和操作在子类中不必定义，子类中还可以定义自己的属性和操作。

如果一个子类只从一个父类得到继承，叫做“单重继承”；如果一个子类有两个或更多个父类，则称为“多重继承”。

5. 多态

不同的对象收到同一消息可以产生完全不同的结果，这一现象叫做多态。在使用多态的时候，用户可以发送一个通用的消息，而实现的细节则由接收对象自行决定，这样，把具有通用功能的消息存放在高层次，而把不同的实现这一功能的行为放在较低层次，在这些低层次上生成的对象能够给通用消息以不同的响应。

多态有几种布控的形式，Cardelli 和 Wegner 把它分为四类，即



其中，参数多态和包含多态称为通用的多态，过载多态和强制多态称为特定的多态。包含多态在许多语言中都存在，最常见的例子就是子类型化，即一个类型是另一个类型的子类型。过载多态是同一个变量被用来表示不同功能而通过上下文以决定一个名字所代表的功能。

6. 动态绑定

绑定是一个把过程调用和响应调用所需要执行的代码加以结合的过程。在一般程序设计语言中，绑定是在编译时进行的，叫做静态绑定。动态绑定则是在运行时进行的，因此，一个给定的过程调用和代码的结合是到调用发生时才进行的。

动态绑定是和类的继承以及多态相联系的。在继承关系中，子类是父类的一个特例，所以父类对象可以出现的地方，子类对象也可以出现。因此在运行过程中，当一个对象发送消息请求服务时，要根据接收对象的具体情况将请求的操作与实现的方法进行连接，即动态绑定。

7.1.2 典型例题分析

例 1 在多态的几种不同形式中，__(37)__ 多态是一种特定的多态，指同一个名字在不同上下文中可代表不同的含义。(2013 年上半年试题 37)

(37) A. 参数　　B. 包含　　C. 过载　　D. 强制

解析：一般将多态分为通用多态和特殊多态。其中通用多态包括参数多态和包含多态，参数多态利用泛型编程，是发散式的，是静态绑定的，让相同的实现代码应用于不同的场合，看重的是算法的普适性，包含多态利用 OOP，是收敛式的，是动态绑定的，让不同的实现代码应用于相同的场合，看重的是接口与实现的分离度。特殊多态包括强制多态和过载多态，其中强制多态即一种类型的变量在作为参数传递时隐式转换成另一种类型，比如一个整型变量可以匹配浮点型变量的函数参数，过载多态同一个名(操作符、函数名)在不同的上下文中有不同的类型。程序设计语言中基本类型的大多数操作符都是过载多态。所以该题考查的是过载多态。

答案：C

例 2　在面向对象方法中，<u>(37)</u>是父类和子类之间共享数据和方法的机制。子类在原有父类接口的基础上，用适合于自己要求的实现去置换父类中的相应实现，称为<u>(38)</u>。(2016 年上半年试题 37、38)

(37) A．封装　　B．继承　　C．覆盖　　D．多态
(38) A．封装　　B．继承　　C．覆盖　　D．多态

解析：继承是父类和子类之间共享数据和方法的机制。

覆盖是子类的方法覆盖了基类的方法，以实现不同的功能，或者对父类的功能进行扩充。

答案：(37) B　　(38) C

例 3　在面向对象技术中，对象具有以下特性：<u>(37)</u>。

① 清晰的边界　② 良好定义的行为　③ 确定的位置和数量　④ 可扩展性 (2012 年下半年试题 37)

(37) A．②④　　B．①②③④　　C．①②④　　D．①②

解析：对象是类的实例，具有清晰的边界、良好定义的行为和可扩展性。

答案：C

例 4　在面向对象技术中，<u>(38)</u>说明一个对象具有多种形态，<u>(39)</u>定义超类与子类之间的关系。(2012 年下半年试题 38、39)

(38) A．继承　　B．组合　　C．封装　　D．多态
(39) A．继承　　B．组合　　C．封装　　D．多态

解析：在面向对象技术中，不同的对象收到同一消息可以产生完全不同的结果，这一现象叫做多态，这说明一个对象具有多种形态；继承是父类和子类之间共享数据和方法的机制，这是类之间的一种关系。

答案：(38) D　　(39) A

例 5　以下关于封装在软件复用中所充当的角色的叙述，正确的是<u>(38)</u>。(2012 年上半年试题 38)

(38) A. 封装使得其他开发人员不需要知道一个软件组织内部是如何工作的
B. 封装使得软件组织更有效地工作
C. 封装使得软件开发人员不需要编制开发文档
D. 封装使得软件组件开发更加容易

解析：封装就是将抽象得到的数据和行为(或功能)相结合，形成一个有机的整体，也就是将数据与操作数据的源代码进行有机地结合，形成“类”，其中数据和函数都是类的成员。封装的目的是增强安全性和简化编程，使用者不必了解具体的实现细节，而只是要通过外部接口，以特定的访问权限来使用类的成员。

答案：A

例6 在有些程序设计语言中，过程调用和响应调用需执行的代码的绑定直到运行时才进行，这种绑定称为 (39) 。(2012年上半年试题39)

(39) A. 静态绑定　B. 动态绑定　C. 过载绑定　D. 强制绑定

解析：绑定指的是一个方法的调用与方法所在的类(方法主体)关联起来。

静态绑定：在程序执行前方法已经被绑定，此时由编译器或其他连接程序实现。

动态绑定：在运行时根据具体对象的类型进行绑定。若一种语言实现了动态绑定，同时必须提供一些机制，可在运行期间判断对象的类型，并分别调用适当的方法。也就是说，编译器此时依然不知道对象的类型，但方法调用机制能自己去调查，找到正确的方法主体。不同的语言对动态绑定的实现方法是有所区别的。

答案：B

例7 采用面向对象开发方法时，对象是系统运行的基本实体。以下关于对象的叙述中，正确的是 (37) 。(2011年下半年试题37)

(37) A. 对象只能包括数据(属性)
B. 对象只能包括操作(行为)
C. 对象一定有相同的属性和行为
D. 对象通常由对象名、属性和操作三部分组成

解析：在面向对象的系统中，对象是基本运行时的实体，它既包括数据(属性)，也包括作用于数据的操作(行为)。一个对象通常可由对象名、属性和操作三部分组成。

答案：D

例8 一个类是 (38) 。在定义类时，将属性声明为private的目的是 (39) 。(2011年下半年试题38、39)

(38) A. 一组对象的封装
B. 表示一组对象的层次关系
C. 一组对象的实例
D. 一组对象的抽象定义

(39) A. 实现数据隐藏，以免意外更改
B. 操作符重载
C. 实现属性值不可更改
D. 实现属性值对类的所有对象共享

解析：类是现实世界或思维世界中的实体在计算机中的反映，它将数据以及这些数据上的操作封装在一起。类是对象的抽象，而对象是类的具体实例。

在类中，可将数据声明为private、protect和public，声明为private的数据为私有的，

只能被类中的成员函数调用，不能在类外访问，从而实现数据隐藏的目的，可防止意外更改。

答案：(38) D　　(39) A

例 9　在面向对象软件开发中，封装是一种__(42)__技术，其目的是使对象的使用者和生产者分离。(2011 年下半年试题 42)

(42) A. 接口管理　　B. 信息隐藏　　C. 多态　　D. 聚合

解析：封装是面向对象的三大要素之一。在面向对象软件开发中，通过类将数据(属性)和函数(行为)封装在一起，在类中通过设定访问权限，实现对信息的隐藏或公开，隐藏类的内部实现，留下一些接口与外界进行信息交换。

答案：B

例 10　下列关于一个类的静态成员的描述中，不正确的是__(38)__。(2011 年上半年试题 38)

(38) A．类的静态方法只能访问该类的静态数据成员

B．静态数据成员可被该类的所有方法访问

C．该类的对象共享其静态数据成员的值

D．该类的静态数据成员的值不可修改

解析：在面向对象程序设计中，可以通过类的静态成员来解决同一个类的对象之间共享数据的问题。静态数据成员是属于类的，因此，静态数据成员可以在没有任何对象的情况下就可以访问到。类的静态方法可以不以捆绑对象的形式调用，在调用时没有当前对象的信息，所以静态方法只能访问静态成员，不能访问非静态的数据成员和方法。静态数据成员的值可以修改，选项 D 错误。

答案：D

例 11　以下关于面向对象方法中继承的叙述，错误的是__(37)__。(2010 年下半年试题 37)

(37) A．继承是父类和子类之间共享数据和方法的机制

B．继承定义了一种类与类之间的关系

C．继承关系中的子类将拥有父类的全部属性和方法

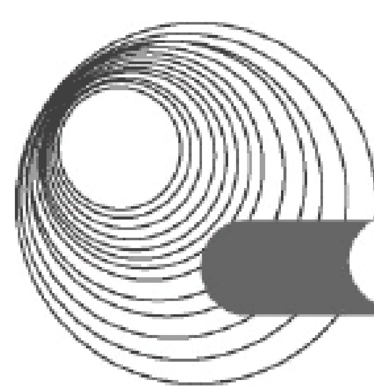
D．继承仅仅允许单重继承，即不允许一个子类有多个父类

解析：面向对象技术中，继承是父类和子类之间共享数据和方法的机制。这是类之间的一种关系，在定义和实现一个类的时候，可以在一个已经存在的类的基础上进行，把这个已经存在的类所定义的内容作为自己的内容，并加入若干新的内容。可以存在多重继承的概念，但不同的程序设计语言可以有自己的规定。

答案：D

例 12　不同的对象收到同一消息可以产生完全不同的结果，这一现象叫做__(38)__。绑定是一个把过程调用和响应调用所需要执行的代码加以结合的过程。在一般的程序设计语言中，绑定在编译时进行，叫做__(39)__；而__(40)__则在运行时进行，即一个给定的过程调用和执行代码的结合直到调用发生时才进行。(2010 年上半年试题 38～40)

(38) A．继承　　B．多态　　C．动态绑定　　D．静态绑定



(39) A．继承　B．多态　C．动态绑定　D．静态绑定

(40) A．继承　B．多态　C．动态绑定　D．静态绑定

解析：在收到消息时，对象要予以响应。不同的对象收到同一消息可以产生完全不同的结果，这一现象叫做多态(Polymorphism)。在使用多态的时候，用户可以发送一个通用的消息，而实现的细节则由接收对象自行决定。这样，同一消息就可以调用不同的方法。绑定是一个把过程调用和响应调用所需要执行的代码加以结合的过程。在一般的程序设计语言中，绑定是在编译时进行的，叫做静态绑定。动态绑定则是在运行时进行的，因此，一个给定的过程调用和代码的结合直到调用发生时才进行。

动态绑定是和类的继承以及多态相联系的。在继承关系中，子类是父类一个特例，所以父类对象可以出现的地方，子类对象也可以出现。因此在运行过程中，当一个对象发送消息请求服务时，要根据接收对象的具体情况将请求的操作与实现的方法进行连接，即动态绑定。

答案：(38) B　(39) D　(40) C

例 13　面向对象的系统中，对象是运行的实体，其组成部分不包括 (37) ；一个类定义了一组大体相似的对象，这些对象共享 (38) 。(2015 年下半年试题 37、38)

(37) A．消息　B．行为(操作)　C．对象名　D．状态

(38) A．属性和状态　B．对象名和状态

C．行为和多重度　D．属性和行为

解析：一个对象通常可以由对象名、属性和方法 3 个部分组成。一个对象把属性和行为封装为一个整体。一个类所包含的方法和数据描述了一组对象的共同行为和属性。

答案：(37) A　(38) D

例 14　对象、类、继承和消息传递是面向对象的 4 个核心概念。其中对象是封装 (37) 的整体。(2015 年上半年试题 37)

(37) A．命名空间　B．要完成任务　C．一组数据　D．数据和行为

解析：封装性是面向对象编程中的三大特性之一，封装性就是把对象的成员属性(数据)和成员方法(方法)结合成一个独立的单位，并尽可能隐蔽对象的内部细节，对外形成一个边界，只保留有限的对外接口，使之与外部发生联系。

答案：D

7.1.3　同步练习

1. ______是指在运行时把过程调用和响应调用所需要执行的代码加以结合。

A．绑定　B．静态绑定　C．动态绑定　D．继承

2. (1) 是指把数据以及操作数据的相关方法组合在同一个单元中，使我们可以把类作为软件中的基本复用单元，提高其内聚度，降低其耦合度。面向对象中的 (2) 机制是对现实世界中遗传现象的模拟，通过该机制基类的属性和方法被遗传给派生类。

(1)～(2) A．封装　B．多态　C．继承　D．变异

3. (1) 以静态或动态的连接方式，为应用程序提供一组可使用的类。 (2) 除了提供

可被应用程序调用的类以外，还基本实现了一个可执行的架构。

(1)~(2)　A．函数库　　B．类库　　C．框架　　D．类属

4．______表示了系统与参与者之间的接口。在每一个用例中，该对象从参与者处收集信息，并将之转换为一种被实体对象和控制对象使用的形式。

A．边界对象　　B．可视化　　C．抽象对象　　D．实体对象

5．在面向对象程序设计中，常常将接口的定义与接口的实现相分离，可定义不同的类实现相同的接口。在程序运行过程中，对该接口的调用可根据实际的对象类型调用其相应的实现。为达到上述目的，面向对象语言须提供______机制。

A．继承和过载　　B．抽象类　　C．继承和重置　　D．对象自身引用

6．下列关于一个类的静态成员的描述中，不正确的是______。

A．该类的对象共享其静态成员变量的值

B．静态成员变量可被该类的所有方法访问

C．该类的静态方法只能访问该类的静态成员变量

D．该类的静态数据成员变量的值不可修改

7.1.4　同步练习参考答案

1．C　　2．(1) A　(2) C

3．(1) B　(2) C　　4．A

5．C　　6．D

7.2　面向对象程序设计

7.2.1　考点辅导

面向对象程序设计(Object Oriented Programming，OOP)的实质是选用一种面向对象程序设计语言(OOPL)，采用对象、类及其相关概念所进行的程序设计。

好的 OOPL 必须至少支持被封装的对象、类和实例概念、类间的继承性和多态。

7.2.1.1　面向对象的好处

面向对象的好处总结如下。

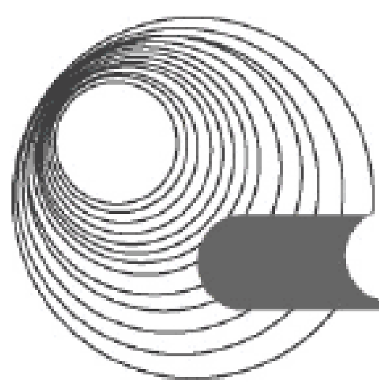
(1)　面向对象技术解决了产品质量和生产率之间的权衡问题。

(2)　面向对象程序设计，特别是继承机制，使得系统具有很高的灵活性和易扩充性。

(3)　面向对象是一个能管理复杂性并增强伸缩性的工具。

(4)　根据面向对象的观点，以现实世界对应物为基础，把某一领域分割成各种对象进行分析与设计，常常比自顶向下进行功能分解的分析及设计更自然合理。

(5)　从概念模型化到分析、设计、编码可以实现无缝传递。



(6) 通过封装进行的信息隐蔽有助于建立安全的系统。

7.2.1.2 面向对象程序设计语言

1. Smalltalk

Smalltalk 并不是一种单纯的程序设计语言，而是反映面向对象程序设计思想的程序设计环境。这个系统在其本身的设计中强调了对象概念的归一性，引入了类、方法、实例等概念和术语，应用了单重继承和动态绑定，成为 OOPLs 发展过程中的一个引人注目的里程碑。

在 Smalltalk-80 中，除了对象之外没有其他形式的数据，对一个对象的唯一操作就是向它发消息。在这种语言中，连类也被看成是对象——类是元类的实例。因此，定义一个类就是向元类发一条消息，请求元类执行它的 new 来生成一个实例，也就是生成这个类，而消息中的参数就是关于这个类的说明。Smalltalk 全面支持面向对象的概念。

2. Eiffel

Eiffel 的主要特点是全面的静态类型化、有大量的开发工具、支持多继承。Eiffel 也全面支持面向对象的概念。

3. C++

C++语言是一种面向对象的强类型语言，由 AT&T 的 Bell 实验室于 1980 年推出。

C++语言是 C 语言的一个向上兼容的扩充，而不是一种新语言。C++是一种支持多范型的程序设计语言，它既支持面向对象的程序设计，也支持面向过程的程序设计。

C++支持基本的面向对象概念，包括对象、类、方法、消息、子类和继承。C++完全支持多继承，并且通过使用 try/throw/catch 模式提供了一个完整的异常处理机制。它同时支持静态类型和动态类型，也完全支持多继承，不提供自动的无用存储单元收集，这必须通过程序员来实现，或者通过编程环境提供合适的代码库来予以支持。

4. Java

Java 语言起源于 Oak 语言，Oak 语言被设计成能运行在设备的嵌入式芯片上。

Java 编译成伪代码，这需要一个虚拟机来对其进行解释，Java 的虚拟机在几乎每一种平台上都可以运行。这实质上使得开发是与机器独立无关的，并且提供了通用的可移植性。

Java 把类的概念和接口的概念区分开来，并试图通过只允许接口的多继承来克服多继承的危险。

Java 的异常处理机制与 C++的 try/throw/catch 相类似，但更加严密。在 Java 中，通过声明轻型线程来处理并发性，这些线程通过副作用和同步协议进行通信。

Java Beans 是组件，即类及其所需资源的集合，它们主要被设计用来提供定制的 GUI 小配件。

Java 中关于面向对象概念的术语有对象、类、方法、实例变量、消息、子类和继承。

7.2.1.3 面向对象程序设计语言中的 OOP 机制

1. 类

类具有实例化功能，包括实例生成和实例消除完成。类的实例化决定了类及其实例具

有下面的特征。

(1) 同一个类的不同实例具有相同的数据结构，承受的是同一方法集合所定义的操作，因而具有规律相同的行为。

(2) 同一个类的不同实例可以持有不同的值，因而可以具有不同的状态。

(3) 实例的初始状态(初值)可以在实例化时确定。

2. 继承和类层次结构

孤立的类只能描述实体集合的特征同一性，而客观世界中实体集合的划分还通常要考虑与实体特征方法有关联的相似性。在 OOP 中使用继承机制解决这一问题。

当执行一个子类的实例生成方法时，首先在类层次结构中从该子类沿继承路径上溯至它的一个基类，然后自顶向下地执行该子类的所有父类的实例生产方法，最后执行该子类实例生产方法的函数体。当执行一个子类的实例消除方法时，顺序正好与之相反：先执行该子类的实例消除方法，再沿继承路径自底向上地执行该子类所有父类的实例消除方法。

类的实例化过程是一种实例的合成过程，而不仅仅是根据单个类型进行的空间分配、初始化和联编。指导编译程序进行这种合成的，就是类层次结构。

3. 对象、消息传递和方法

对象是类的实例。尽管对象的表示在形式上与一般数据结构十分相似，但是它们之间存在本质区别：对象之间通过消息传递方式进行通信。

消息传递原是一种与通信有关的概念，OOP 使得对象具有交互能力的主要模型就是消息传递模型。对象被看成用传递消息的方式互相联系的通信实体，它们既可以接收，也可以拒绝外界发来的消息。一般情况下，对象接收它能够识别的消息，拒绝不能识别的消息。

发送一条消息至少应给出一个对象的名字和要发送给这个对象的那条消息的名字。通常，消息的名字就是这个对象中外界可知的某个方法的名字。在消息中，还经常有一组参数，将外界的有关信息传递给这个对象。

对于一个类来说，它关于方法界面的定义规定了实例的消息传递协议，而它本身则决定了消息传递到合法范围。

4. 对象自身引用

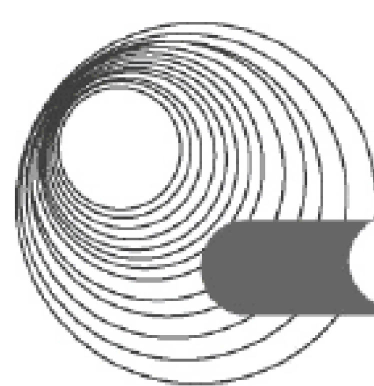
对象自身引用是 OOPLs 中的一种特有结构。这种结构在不同的 OOPLs 中有不同的名称，在 C++和 Java 中称为 this，在 Smalltalk-80、Object-C 和其他一些 OOPLs 中则称为 self。

对象自身引用的值和类型分别扮演了两种意义的角色：对象自身引用的值使得方法体中引用的成员名与特定的对象相关，对象自身引用的类型则决定了方法体被实际共享的范围。

对象自身引用机制使得在进行方法的设计和实现时并不需要考虑与对象联系的细节，而是从更高一级的抽象层次，也就是类的角度来设计同类型对象的行为特征，从而使得方法在一个类及其子类的范围内具有共性。

5. 重置

重置的基本思想是：通过一种动态绑定机制的支持，使得子类在继承父类界面定义的前提下，用适合于自己要求的实现去置换父类中的相应实现。



在 OOPLs 中，重置机制有相应的语法供开发人员选择使用。在 C++语言中，通过虚拟函数(Virtual Function)的定义来进行重置的声明，通过虚拟函数跳转表(Virtual Functions Jump Tables，VTBL)结构来实现重置方法体的动态绑定。在 Java 语言中，通过抽象方法(Abstract Method)来进行重置的声明，通过方法查找(Method Lookup)实现重置方法体的动态绑定。

6. 类属类

类属是程序设计语言中普遍注重的一种参数多态机制。类属类可以看作类的模板。一个类属类是关于一组类的一个特性抽象，它强调的是这些类的成员特征中与具体类型无关的那些部分，而与具体类型相关的部分则用变元来表示。这就使得对类的集合也可以按照特性的相似性再次进行划分。类属类的一个重要作用，就是对类库的建设提供强有力的支持。

7. 无实例的类

要创建无实例的类需要语言的支持。在 C++和 Java 语言中，抽象类就是这样的类。在 C++中通过在类中定义纯虚拟函数来创建一个抽象类，在 Java 中通过在类中定义抽象方法来创建一个抽象类，或者直接将一个类声明为抽象类。

7.2.1.4 面向对象的程序

类库是一种预先定义的程序库，可以由开发人员自己扩充。

类库以程序模块的形式，按照类层次结构把一组类的定义和实现组织在一起。通常，这一组类预先提供的是一些低层功能，这些功能覆盖了传统例程库所提供的功能。

开发人员可以利用类库中的类，其方式与使用语言中的基本类型完全相同。

7.2.2 典型例题分析

例 1 许多程序设计语言规定，程序中的数据都必须具有类型，其作用不包括(20)。(2009 年下半年试题 20)

(20) A．便于为数据合理分配存储单元

B．便于对参与表达式计算的数据对象进行检查

C．便于定义动态数据结构

D．便于规定数据对象的取值范围及能够进行的运算

解析：不同程序语言所提供的数据类型不尽相同。数据是程序操作的对象，具有名称、类型、存储类、作用域和生存期等属性，使用时要为它分配内存空间。

数据名称可通过标识符命名：类型说明数据占用内存的大小和存放形式，存储类说明数据在内存中的位置和生存期；作用域说明数据可以使用的范围；生存期说明数据占用内存的时间。数据从不同角度可分成不同的类别。按数据的作用域大小，可分为全局量和局部量；按生存期可分为自动生存期、静态生存期和动态生成期；按程序运行时数据的值是否能改变可分为常量和变量。

答案：B

例 2　以下关于 C/C++语言指针变量的叙述中，正确的是(21)。(2009 年下半年试题 21)

(21) A．指针变量可以是全局变量也可以是局部变量

B．必须为指针变量与指针所指向的变量分配相同大小的存储空间

C．对指针变量进行算术运算是没有意义的

D．指针变量必须由动态产生的数据对象来赋值

解析： 存放地址的变量称为指针变量。指针变量是一种特殊的变量，它不同于一般的变量，一般变量存放的是数据本身，而指针变量存放的是数据的地址。A 选项显然是正确的。对于 B 选项，指针变量和指针所指向的变量存放的内容是不一样的，只要分配够用就行了，不需要分配一样大小的存储空间。对于 C 选项，指针变量加 1 就相当于地址加 1，是有意义的。另外，指针变量可以静态地定义。

答案： A

例 3　一个类可以具有多个同名而参数类型列表不同的方法，被称为方法 (39) 。(2015 年下半年试题 39)

(39) A．重载　　B．调用　　C．重置　　D．标记

解析： 简单地说，重载就是函数或者方法有同样的名称，但是参数列表不相同的情形，这样的同名不同参数的函数或者方法之间，互相称之为重载函数或者方法。重置，则是在子类中重新定义父类中已经定义的方法，这是一种动态绑定机制。要注意区分重载和重置。

答案： A

例 4　面向对象 (38) 选择合适的面向对象程序设计语言，将程序组织为相互协作的对象集合，每个对象表示某个类的实例，类通过继承等关系进行组织。(2015 年上半年试题 38)

(38) A．分析　　B．设计　　C．程序设计　　D．测试

解析： 面向对象分析的目的是确定系统的功能、性能要求。面向对象设计是设计分析模型和实现相应的源代码。面向对象程序设计的实质是选用一种面向对象程序设计语言，采用类、对象及其相关概念所进行的程序设计。

答案： C

7.2.3　同步练习

1. 在选择某种面向对象语言进行软件开发时，不需要着重考虑的因素是，该语言_____。

A．将来是否能够占据市场主导地位　　B．类库是否丰富

C．开发环境是否成熟　　D．是否支持全局变量和全局函数的定义

2. 在面向对象的语言中，_____。

A．类的实例化是指对类的实例分配存储空间

B．每个类都必须创建一个实例

C．每个类只能创建一个实例

D．类的实例化是指对类进行初始化

3. 面向对象程序设计语言为_____提供支持。

A．面向对象用例设计阶段　　B．面向对象分析阶段
C．面向对象需求分析阶段　　D．面向对象实现阶段

4. 下面关于面向对象的描述，正确的是_____。

A．针对接口编程，而不是针对实现编程
B．针对实现编程，而不是针对接口编程
C．接口与实现不可分割
D．优先使用继承而非组合

5. 重置的基本思想是通过_____机制的支持，使得子类在继承父类界面定义的前提下，用适合于自己要求的实现去置换父类中的相应实现。

A．静态绑定　　B．对象引用　　C．类型匹配　　D．动态绑定

6. 在面向对象技术中，类属是一种 (1) 机制。一个类属类是关于一组类的一个特性抽象，它强调的是这些类的成员特征中与 (2) 的那些部分，而用变元来表示与(3)的那些部分。

(1) A．包含多态　　B．参数多态　　C．过载多态　　D．强制多态

(2)～(3) A．具体对象无关　　B．具体类型无关
C．具体对象相关　　D．具体类型相关

7.2.4　同步练习参考答案

1．D　　2．A
3．D　　4．A
5．D　　6．(1) B　(2) B　(3) D

7.3　面向对象开发技术

7.3.1　考点辅导

7.3.1.1　面向对象分析

面向对象分析的目的是为了获得对应用问题的理解。理解的目的是确定系统的功能、性能要求。

面向对象分析包含 5 个活动，即认定对象、组织对象、描述对象间的相互作用、定义对象的操作和定义对象的内部信息。

1. 认定对象

在应用领域中，按自然存在的实体确立对象。在定义域中，首先将自然存在的“名词”作为一个对象，这通常是研究问题、定义域实体的良好开始。通过实体间的关系寻找对象常常没有问题，而困难在于寻找(选择)系统关心的实质性对象，实质性对象是系统稳定性的基础。

2. 组织对象

分析对象间的关系，将相关对象抽象成类，其目的是为了简化关联对象，利用类的继承性建立具有继承性层次的类结构。抽象类时可从对象间的操作或一个对象是另一个对象的一部分来考虑，如房子由门和窗构成，门和窗是房子类的子类。由对象抽象类，通过相关类的继承构造类层次，所以说系统的行为和信息间的分析过程是一种迭代表征过程。

3. 描述对象间的相互作用

描述出各对象在应用系统中的关系，如一个对象是另一个对象的一部分、一个对象与其他对象间的通信关系等。这样可以完整地描述每个对象的环境，由一个对象解释另一个对象，以及一个对象如何生成另一个对象，最后得到对象的界面描述。

4. 定义对象的操作

当考虑对象的界面时，自然要考虑对象的操作。其操作有从对象直接标识的简单操作，如创建、增加和删除等；也有更复杂的操作，如将几个对象的信息连接起来。一般而言，应避免对象太复杂，当连接的对象很复杂时，可将其标识为新对象。当确定对象的操作后，再定义对象的内部。对象内部定义包括其内部数据信息、信息存储方法、继承关系以及可能生成的实例数等属性。

7.3.1.2　面向对象设计

面向对象设计的含义是设计分析模型和实现相应源代码，在目标代码环境中这种源代码可被执行。设计期间必须充分考虑系统的稳定性，这会影响系统的结构。

对象标识期间的目标是分析对象，设计过程也是发现对象的过程，称之为再处理。对象可以用预先开发的源代码实现，称这样的部分为构件。

7.3.1.3　面向对象测试

就测试而言，用面向对象方法开发的系统测试与其他方法开发的系统测试没有什么不同，在所有开发系统中都是根据规范说明来验证系统设计的正确性。程序调试步骤是从最底层开始，从单元测试、综合测试到系统测试。

一般来说，对面向对象软件的测试可分为下列 4 个层次进行。

(1) 算法层。测试类中定义的每个方法，基本上相当于传统软件测试中的单元测试。

(2) 类层。测试封装在同一个类中的所有方法与属性之间的相互作用。在面向对象软件中类是基本模块，因此可以认为这是面向对象测试中所特有的模块测试。

(3) 模板层。测试一组协同工作的类之间的相互作用。大体上相当于传统软件测试中的集成测试，但是也有面向对象软件的特点(如对象之间通过发送消息相互作用)。

(4) 系统层。把各个子系统组装成完整的面向对象软件系统，在组装过程中同时进行测试。

7.3.2　典型例题分析

例 1　业务用例和参与者一起描述 (46) ，而业务对象模型描述 (47) 。(2011 年下半年

试题46、47)

(46) A. 工作过程中的静态元素　　B. 工作过程中的动态元素
C. 工作过程中的逻辑视图　　D. 组织支持的业务过程

(47) A. 业务结构
B. 结构元素如何完成业务用例
C. 业务结构以及元素如何完成业务用例
D. 组织支持的业务过程

解析：业务用例模型描述一个业务的流程及其与外部各方(如客户和合作伙伴)之间的交互。业务用例模型描述的是业务范围，与系统用例模型讲述的系统范围是不同的。

业务对象模型是描述业务用例实现的对象模型。业务对象模型是一种根据职责、可交付工件和协作行为来说明业务流程的方法。不建立成业务对象模型意味着存在一定风险，因为开发人员只是肤浅地注意业务的进行方式，用自己已有的观念思考，即在缺乏业务流程观念的前提下设计和创建软件，其结果可能是构建出来的系统无法满足业务需要。

答案：(46) D　(47) C

例2　面向对象分析的第一步是__(37)__。(2011年上半年试题37)

(37) A. 定义服务　　B. 确定附加的系统约束
C. 确定问题域　　D. 定义类和对象

解析：面向对象分析(OOA)需要将真实世界进行抽象，通过问题的叙述，将真实世界系统加以描述。分析的目的是为了构造一个系统属性和系统行为的模型，该模型是根据对象和对象之间的关系、动态控制和功能转移来确定的。OOA应该包含以下几个步骤。

① 分析问题域，建立用例模型。
② 发现和定义对象和类。
③ 识别对象的内部特征。
④ 识别对象的外部联系。
⑤ 识别对象之间的交互。

面向对象分析的基础就是问题域以及用户的需求，研究问题域和系统需求的主要目的是通过对问题域的深入研究，建立一个能够满足用户需求的系统模型。面向对象分析的第一步就是分析用户需求，确定问题域，在此基础上建立用例模型。

答案：C

例3　开-闭原则(Open-Closed Principle，OCP)是面向对象的可复用设计的基石。开-闭原则是指一个软件实体应当对__(37)__开放，对__(38)__关闭；里氏代换原则(Liskov Substitution Principle，LSP)是指任何__(39)__可以出现的地方，__(40)__一定可以出现。依赖倒转原则(Dependence Inversion Principle，DIP)就是要依赖于__(41)__而不依赖于__(42)__，或者说要针对接口编程，不要针对实现编程。(2010年下半年试题37～42)

(37) A. 修改　B. 扩展　C. 分析　D. 设计
(38) A. 修改　B. 扩展　C. 分析　D. 设计
(39) A. 变量　B. 常量　C. 基类对象　D. 子类对象

(40) A．变量　　B．常量　　C．基类对象　　D．子类对象
(41) A．程序设计语言　　B．建模语言
　　C．实现　　D．抽象
(42) A．程序设计语言　　B．建模语言
　　C．实现　　D．抽象

解析：开-闭原则可以说是面向对象设计的核心所在。开-闭原则的两个重要特点是“对扩展开放，对修改关闭”，即允许对程序作出扩展(以扩展的方式响应需求的变化)，但拒绝对程序作出修改(即修改之前运行良好的程序)。实现“开-闭原则”的重要机制就是“抽象”与“多态”。通过对“变化”进行抽象隔离，使程序具有更好的可扩展性与可维护性。

里氏代换原则在实现继承时，子类(Subtype)必须能替换掉它们的基类(Base Type)。如果一个软件实体使用的是基类，那么也一定适用于子类；但反过来的代换不成立。

依赖倒转原则是指在进行业务设计时，与特定业务有关的依赖关系应该尽量依赖接口和抽象类，而不是依赖具体类。具体类只负责相关业务的实现，修改具体类不影响与特定业务有关的依赖关系。

答案：(37) B　(38) A　(39) C　(40) D　(41) D　(42) C

例 4　采用面向对象方法开发软件的过程中，抽取和整理用户需求并建立问题域精确模型的过程叫 (16) 。(2010 年上半年试题 16)

(16) A．面向对象测试　　B．面向对象实现
　　C．面向对象设计　　D．面向对象分析

解析：采用面向对象的软件开发，通常有面向对象分析、面向对象设计、面向对象实现。面向对象分析是为了获得对应用问题的理解，其主要任务是抽取和整理用户需求并建立问题域精确模型。面向对象设计是采用协作的对象、对象的属性和方法说明软件解决方案的一种方式，强调的是定义软件对象和这些软件对象如何协作来满足需求，延续了面向对象分析。面向对象实现主要强调采用面向对象程序设计语言实现系统。面向对象测试是根据规范说明来验证系统设计的正确性。

答案：D

例 5　(41) 不是面向对象分析阶段需要完成的。(2010 上半年试题 41)

(41) A．认定对象　　B．组织对象
　　C．实现对象及其相互关系　　D．描述对象间的相互作用

解析：面向对象分析包含 5 个活动，即认定对象、组织对象、描述对象间的相互作用、定义对象的操作、定义对象的内部信息。

实现对象及其相互关系应该归入系统的实现阶段，不属于分析阶段的任务。

答案：C

例 6　以下关于面向对象设计的叙述中，错误的是 (42) 。(2010 年上半年试题 42)

(42) A．面向对象设计应在面向对象分析之前，因为只有产生了设计结果才可对其进行分析
　　B．面向对象设计与面向对象分析是面向对象软件过程中两个重要的阶段

C．面向对象设计应该依赖于面向对象分析的结果

D．面向对象设计产生的结果在形式上可以与面向对象分析产生的结果类似，例如都可以使用 UML 表达

解析：面向对象分析与设计是面向对象软件开发过程中的两个重要阶段，面向对象分析产生分析模型，该分析模型可以使用 UML 表达，面向对象设计以分析模型为基础，继续对分析模型进行精化，得到设计模型，其表达仍然可以采用 UML 建模语言。

答案：A

例 7　以下关于面向对象分析的叙述中，错误的是(37)。(2009 年下半年试题 37)

(37) A．面向对象分析看重分析问题域和系统责任

B．面向对象分析需要考虑系统的测试问题

C．面向对象分析忽略与系统实现有关的问题

D．面向对象分析建立独立于实现的系统分析模型

解析：面向对象分析的目的是为了获得对应用问题的理解。理解的目的是确定系统的功能和性能要求。面向对象分析侧重于理解问题和理解解决方案，不注重考虑系统的实现，也谈不上考虑系统的测试问题，显然 B 选项是错误的。

答案：B

例 8　下面关于面向对象分析与面向对象设计的说法中，不正确的是(37)。(2009 年上半年试题 37)

(37) A．面向对象分析侧重于理解问题

B．面向对象设计侧重于理解解决方案

C．面向对象分析描述软件要做什么

D．面向对象设计一般不关注技术和实现层面的细节

解析：面向对象开发过程的核心是面向对象分析(OOA)和面向对象设计(OOD)两个阶段，但二者的界限比较模糊。OOA 是分析使用实例，提取用户需求，建立问题域逻辑模型的过程；OOD 是建立面向对象的求解域模型的过程。从 OOA 到 OOD 实际是一个多次反复、逐步迭代模型的过程。

面向对象分析是采用面向对象思路进行需求分析建模的过程。面向对象的分析模型主要有用例模型、类/对象模型、对象-关系模型和对象-行为模型等。类/对象模型描述系统所涉及的全部类/对象，每个类/对象都通过属性、操作、协作者进一步描述；对象-关系模型描述对象之间的静态关系，同时定义了系统所有重要的消息路径，它也可以涉及对象的属性、操作、协作者；对象-行为模型描述了系统动态行为。

面向对象设计直接继承面向对象分析阶段的类图和交互图等分析结果，然后确定每个类内部的数据和方法，以及每个方法的处理算法、过程和接口等。因此 D 选项是错误的。

答案：D

例 9　在面向对象分析与设计中，(38)是应用领域中的核心类，一般用于保存系统中的信息以及提供针对这些信息的相关处理行为；(39)是系统内对象和系统外参与者的联系媒介；(40)主要是协调上述两种类对象之间的交互。(2009 年上半年试题 38～40)

(38)～(40) A．控制类　　B．边界类　　C．实体类　　D．软件类

解析：实体类(Entity Class)是应用领域中的核心类，一般是从现实世界中的实体对象归纳和抽象出来的，用于长期保存系统中的信息，以及提供针对这些信息的相关处理行为。一般情况下，实体类的对象实例和应用系统本身有着相同的生命周期。

边界类(Boundary Class)是系统内的对象和系统外的参与者的联系媒体，外界的消息只能通过边界类的对象实例发送给系统。

控制类(Control Class)是实体类和边界类之间的润滑剂，是从控制对象中归纳和抽象出来的，用于协调系统内边界类和实体类之间的交互。

答案：(38) C　(39) B　(40) A

7.3.3　同步练习

面向对象分析与设计中的 (1) 是指一个模块在可扩展性方面应该是开放的，而在更改性方面应该是封闭的；而(2)是指子类应当可以替换父类并出现在父类能够出现的任何地方。

(1)～(2) A．开放封闭原则　　B．替换原则

C．依赖原则　　D．单一职责原则

7.3.4　同步练习参考答案

(1) A　　(2) C

7.4　面向对象的分析与设计方法

7.4.1　考点辅导

7.4.1.1　Peter Coad 和 Edward Yourdon 的 OOA 和 OOD 法

1．OOA

OOA 模型由下列 5 个层次和 5 个活动组成。

(1)　5 个层次：主题层、对象类层、结构层、属性层、服务层。

(2)　5 个活动：标志对象类、标志结构、定义主题、定义属性、定义服务。

在这种方法中定义了两种对象类之间的结构，一种称为分类结构，另一种称为组装结构。

2．OOD

OOA 中的 5 个层次和 5 个活动继续贯穿在 OOD 过程中。OOD 模型由 4 个部分和 4 个活动组成。

4 个活动是设计问题域部件、设计人机交互部件、设计任务管理部件、设计数据管理

部件。

7.4.1.2 Booch 的 OOD 法

Booch 认为软件开发是一个螺旋上升的过程，在螺旋上升的每个周期中有以下步骤。

(1) 标识类和对象。

(2) 确定它们的含义。

(3) 标志它们之间的关系。

(4) 说明每一个类的界面和实现。

除了类图、对象图、模块图、进程图外，Booch 的 OOD 中还使用了两种动态描述图，一种是刻画特定类实例的状态转换图，另一种是描述对象间事件变化的时序图。

7.4.1.3 OMT 法

对象建模技术(OMT)定义了 3 种模型，即对象模型、动态模型和功能模型。OMT 方法有 4 个步骤，即分析、系统设计、对象设计和实现。

1. 对象模型、动态模型和功能模型

1) 对象模型

OMT 的对象模型中除了对象、类、继承外，还有一些其他的概念。

- 链和关联。链表示实例对象间的物理或概念上的连接。关联描述具有公共结构和公共语义的一组链。
- 泛化。泛化是一个类与它的一个或多个细化类之间的关系，即一般与特殊的关系。被细化的类称为父类，每个细化的类称为子类，子类可以继承父类的特性。
- 聚集。聚集是一种整体与部分的关系，在这种关系中表示整体的对象与表示部分的对象关联。
- 模块。模块是组和类、关联和泛化的一种逻辑结构，模块给出了某个主题的视图。

2) 动态模型

动态模型描述与时间和操作顺序有关的系统特征，包括激发事件、事件序列、确定事件先后关系以及事件和状态的组织。

3) 功能模型

功能模型描述与值的变换有关的系统特征，包括功能、映射、约束和函数依赖。

对象模型、动态模型和功能模型之间具有下述关系。

(1) 与功能模型的关系。对象模型展示了功能模型中的动作者、数据存储和流的结构，动态模型展示了执行加工的顺序。

(2) 与对象模型的关系。功能模型展示了类上的操作和每个操作的变量，因此它也表示了类之间的“供应者—客户”关系；动态模型展示了每个对象的状态以及它接收事件和改变状态时所执行的操作。

(3) 与动态模型的关系。功能模型展示了动态模型中未定义的不可分解的动作和活动的定义，对象模型展示了是谁改变了状态和承受了操作。

2. OMT 的步骤

下面介绍 OMT 的步骤。

(1) 分析。目的是建立可理解的现实世界模型。

(2) 系统设计。确定整个系统的体系结构，形成求解问题和建立解答的高层次策略。

(3) 对象设计。建立基于分析模型的设计模型，并考虑实现的细节，设计人员根据系统设计期间建立的策略把实现细节加入到设计模型中。

(4) 实现。将对象设计阶段开发的对象类及其关系转换成特定的程序设计语言、数据库或硬件。

7.4.1.4　UML 概述

统一建模语言(UML)是面向对象软件的标准化建模语言。UML 由 3 个要素构成，即 UML 的基本构造块、支配这些构造块如何放置在一起的规则和运用于整个语言的一些公共机制。UML 的词汇表包含 3 种构造块，即事物、关系和图。事物是对模型中最具代表性的成分的抽象，关系把事物结合在一起，图聚集了相关的事物。

1. 事物

事物包括结构事物、行为事物、分组事物和注释事物。

(1) 结构事物(Structural Thing)。结构事物是 UML 模型中的名词。它们通常是模型的静态部分，描述概念或物理元素。结构事物包括类(Class)、接口(Interface)、协作(Collaboration)、用例(Use Case)、主动类(Active Class)、构件(Component)和节点(Node)。

(2) 行为事物(Behavior Thing)。行为事物是 UML 模型的动态部分。它们是模型中的动词，描述了跨越时间和空间的行为。共有两类主要的行为事物，即交互(Interaction)和状态机(State Machine)。

(3) 分组事物(Grouping Thing)。分组事物是 UML 模型的组成部分。它们是一些由模型分解成的“盒子”。在所有的分组事物中，最主要的分组事物是包(Package)。

(4) 注释事物(Annotational Thing)。注释事物是 UML 模型的解释部分。这些注释事物用来描述、说明和标注模型的任何元素。注解(Note)是一种主要的注释事物。注解是一个依附于一个元素或者一组元素之上，并对它进行约束或解释的简单符号。

2. 关系

UML 中有 4 种关系，即依赖、关联、泛化和实现。

(1) 依赖(Dependency)。依赖是两个事物间的语义关系，其中一个事物(独立事物)发生变化会影响另一个事物(依赖事物)的语义。

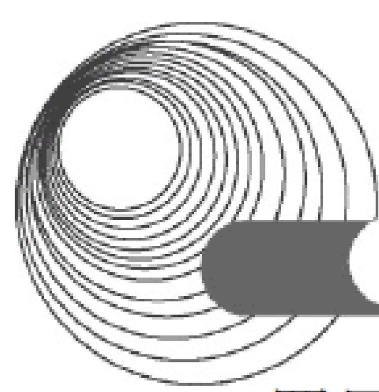
(2) 关联(Association)。关联是一种结构关系，它描述了一组链，链是对象之间的连接。聚集(Aggregation)是一种特殊类型的关联，它描述了整体和部分间的结构关系。

(3) 泛化(Generalization)。泛化是一种特殊一般关系，特殊元素(子元素)的对象可替代一般元素(父元素)的对象。用这种方法子元素共享了父元素的结构和行为。

(4) 实现(Realization)。实现是类元之间的语义关系，其中一个类元指定了由另一个类元保证执行的契约。在两种地方要遇到实现关系：一种是在接口和实现它们的类或构件之间；另一种是在用例和实现它们的协作之间。

3. UML 中的图

UML 提供的图包括类图、对象图、用例图、交互图、状态图、活动图、构件图和部



署图。

(1) 类图(Class Diagram)展现了一组对象、接口、协作及其关系。类图给出系统的静态设计视图。包含主动类的类图给出了系统的静态进程视图。

(2) 对象图(Object Diagram)展现了一组对象及其关系。对象图描述了在类图中所建立的事物的实例的静态快照。对象图一般包括对象和链。

(3) 用例图(Use Case Diagram)展现了一组用例、参与者(Actor)及其关系。

(4) 序列图、通信图、交互概览图和时序图均被称为交互图，它们用于对系统的动态方面进行建模。

(5) 状态图(State Diagram)展现了一个状态机，它由状态、转换、事件和活动组成。状态图关注系统的动态视图，它对于接口、类和协作的行为建模尤为重要，强调对象行为的事件顺序。

(6) 活动图(Activity Diagram)是一种特殊的状态图，它展现了在系统内从一个活动到另一个活动的流程。活动图专注于系统的动态视图，它对于系统的功能建模特别重要，并强调对象间的控制流程。

(7) 构件图(Component Diagram)展现了一组构件之间的组织和依赖。构件图专注于系统的静态实现视图。它与类图相关，通常把构件映射为一个或多个类、接口或协作。

(8) 部署图(Deployment Diagram)展现了运行处理节点以及其中构件的配置。部署图给出了体系结构的静态实施视图。它与构件图相关，通常一个节点包含一个或多个构件。

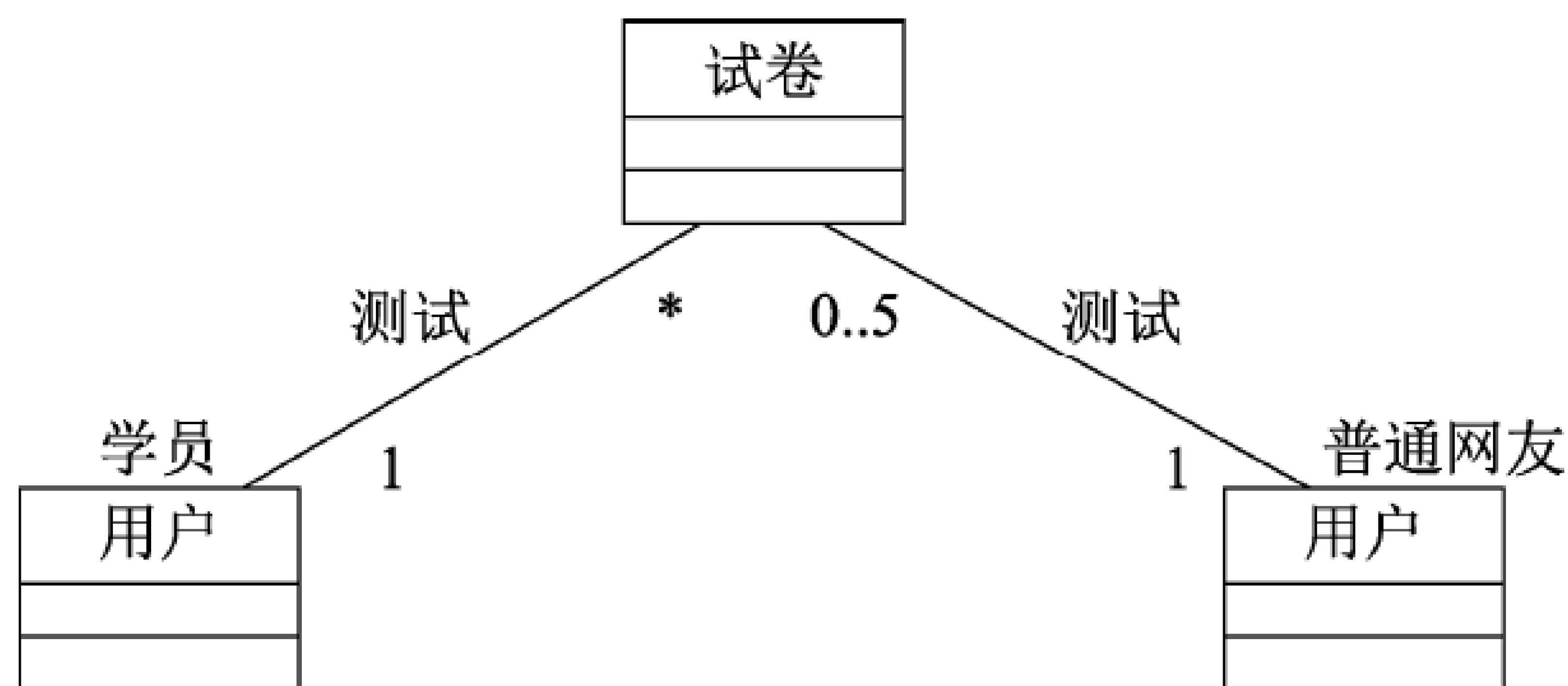
7.4.2 典型例题分析

例 1 UML 中关联是一个结构关系，描述了一组链。两个类之间 (40) 关联。(2016 年上半年试题 40)

(40) A．不能有多个　　B．可以有多个由不同角色标识的

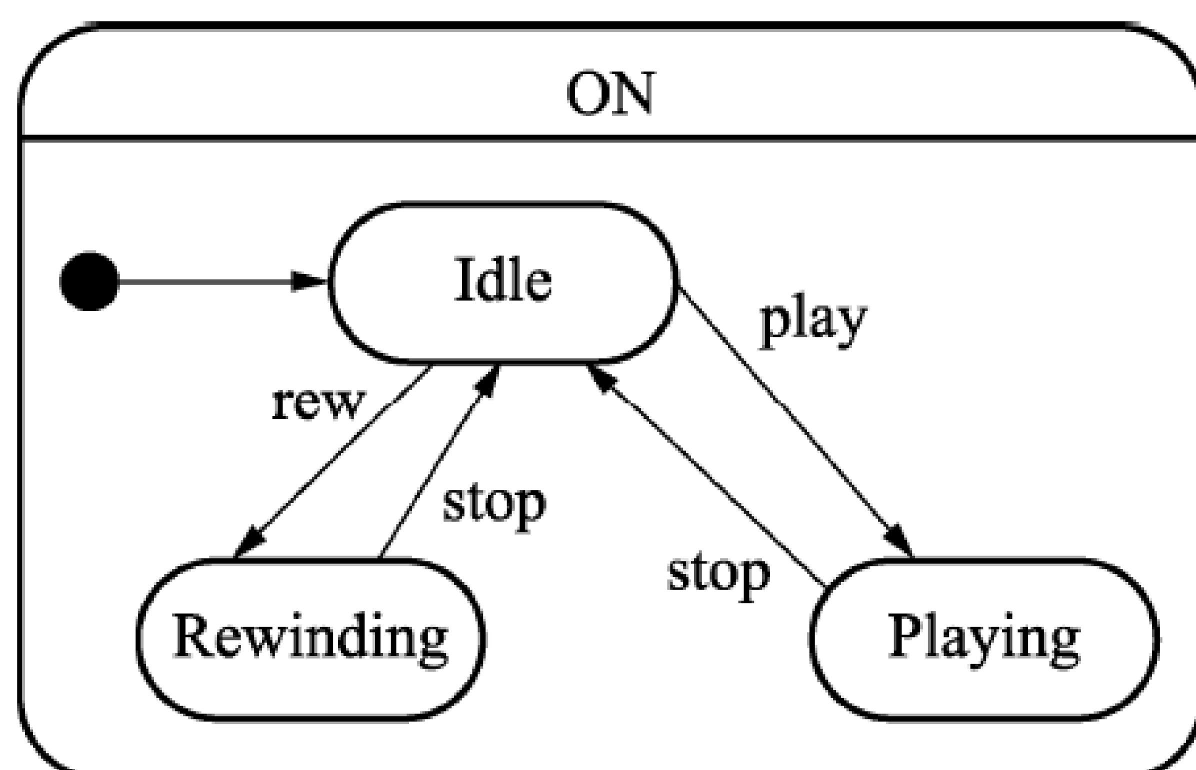
C．可以有任意多个　　D．多个关联必须聚合成一个

解析：两个类之间可以由不同角色标识存在多个关联，如下图所示。



答案：B

例 2 以下关于 UML 状态图的叙述中，不正确的是 (41) 。对下图的描述正确的是 (42) 。(2013 年上半年试题 41、42)



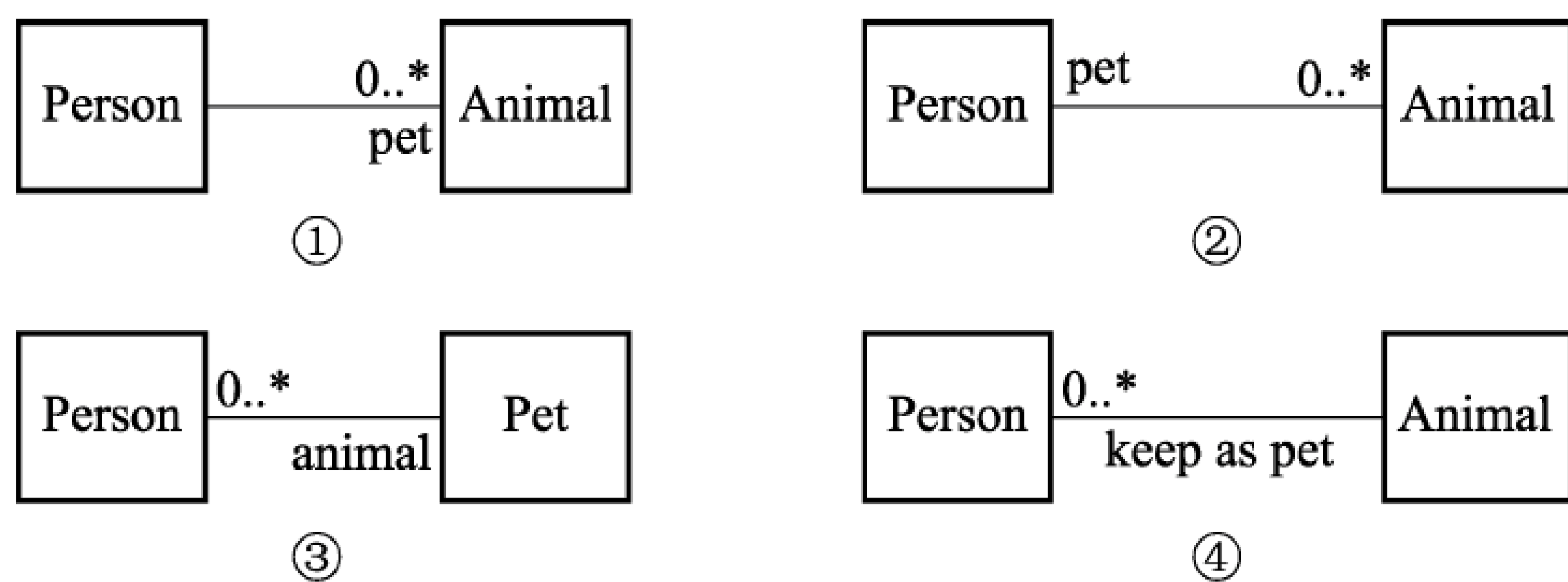
(41) A. 用于描述一个对象在多个用例中的行为
B. 用于某些具有多个状态的对象而不是系统中大多数或全部对象
C. 用于描述多个对象之间的交互
D. 可以用于用户界面或控制对象

(42) A. ON 是一个并发状态
B. 因为此状态图中没有终止(Final)状态，所以此图是无效的
C. play、stop 和 rew 是动作
D. ON 是超状态

解析：状态图展现了一个状态机，它由状态、转换、事件和活动组成。状态图关注系统的动态视图，它对接口、类和协作的行为建模尤为重要，它强调对象行为的事件顺序。状态图通常包含简单状态和组合状态、转换(事件和动作)。可以用状态图对系统的动态方面建模。这些动态方面可以包括出现在系统体系结构的任何视图中的任何一种对象的按事件排序的行为，这些对象包括类(主动类)、接口、构件和节点。所以状态图不表示多个对象之间的交互。根据 ON 状态的内部行为可以发现该状态为超状态。

答案：(41) C　　(42) D

例 3　描述一些人(Person)将动物(Animal)养为宠物(Pet)的是图__(43)__。(2013 年上半年试题 43)

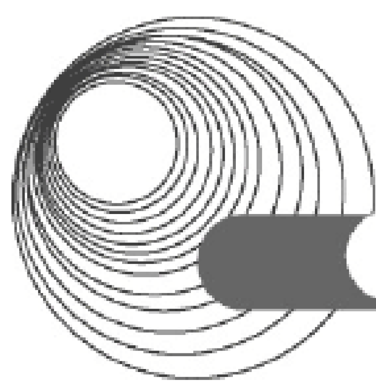


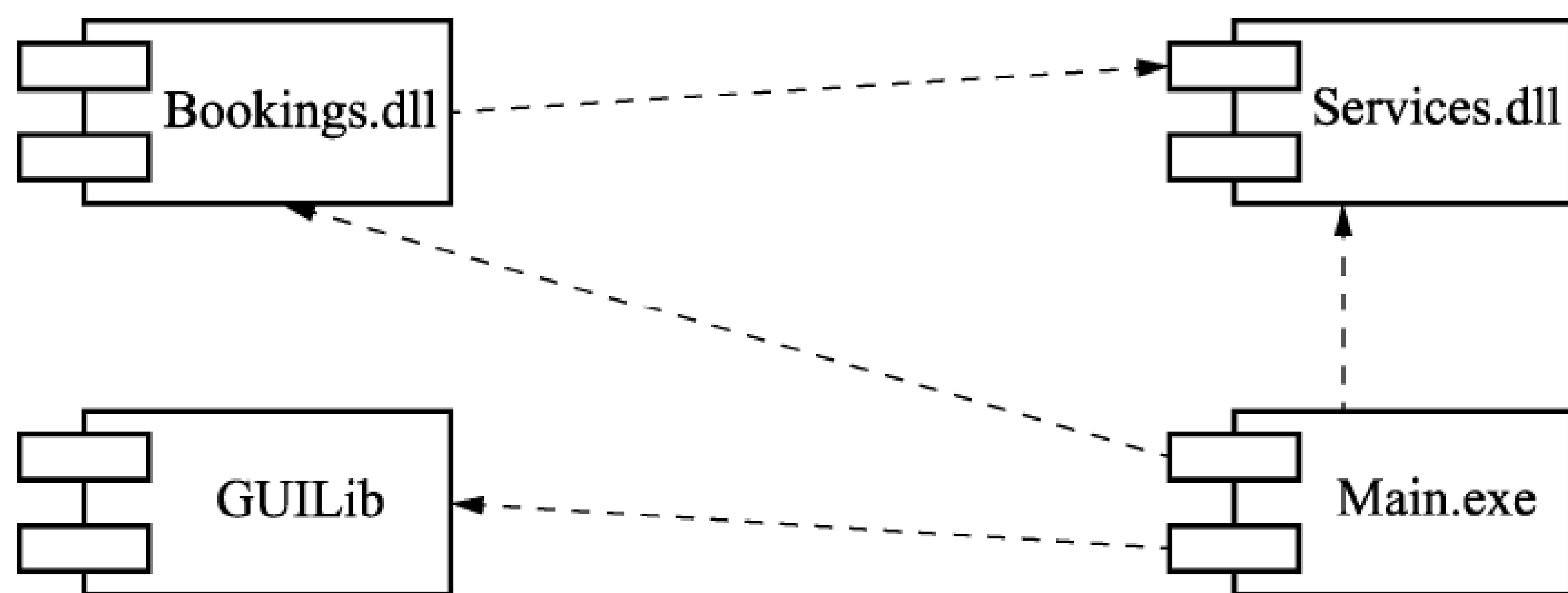
(43) A. ①　　B. ②　　C. ③　　D. ④

解析：该题考查 UML 中类图的表示，由 Person 类和 Animal 类之间一对多的关系以及类图的画法可知选项为 A。

答案：A

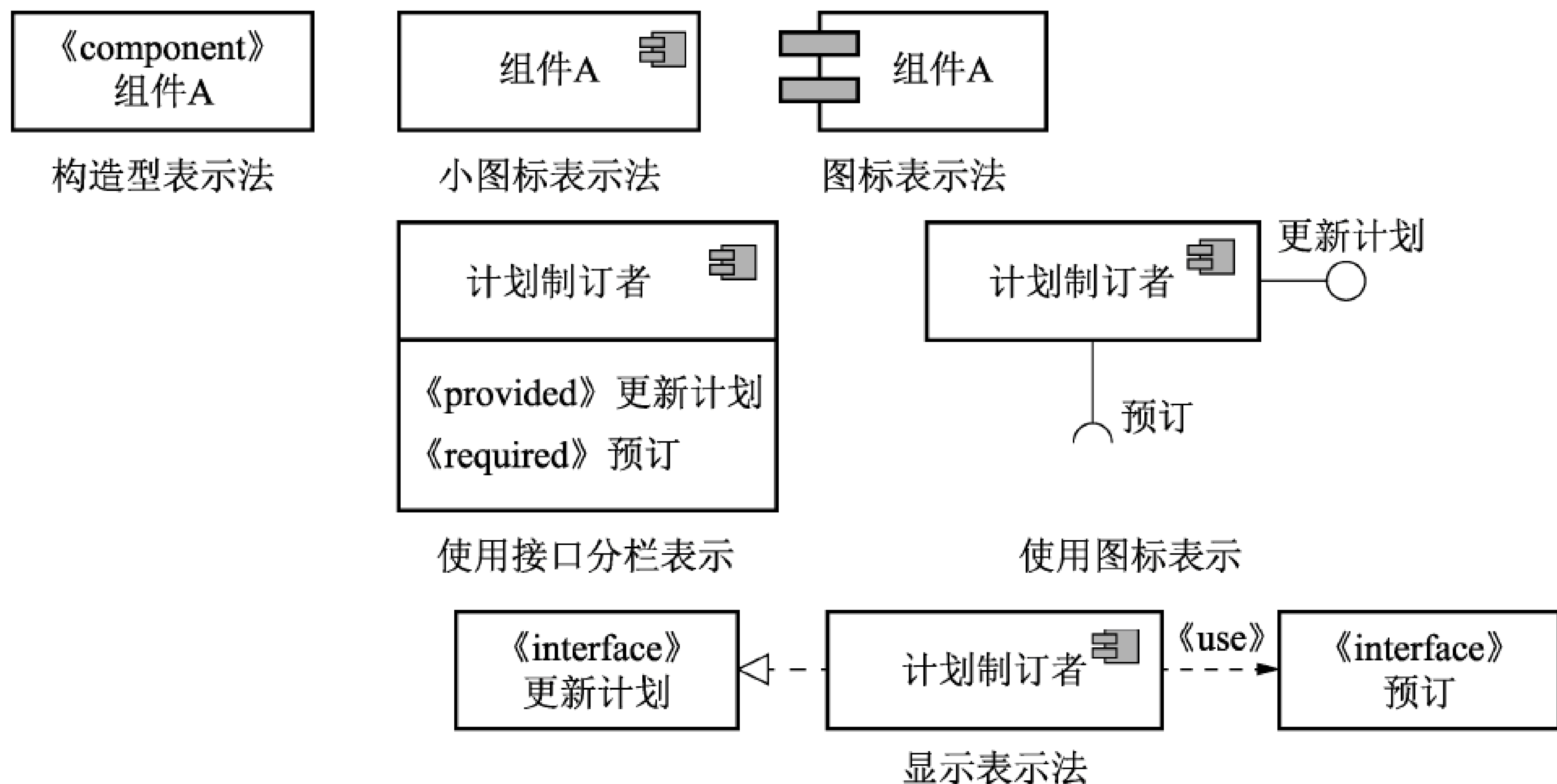
例 4　下图所示为 UML__(43)__。(2015 年上半年试题 43)。





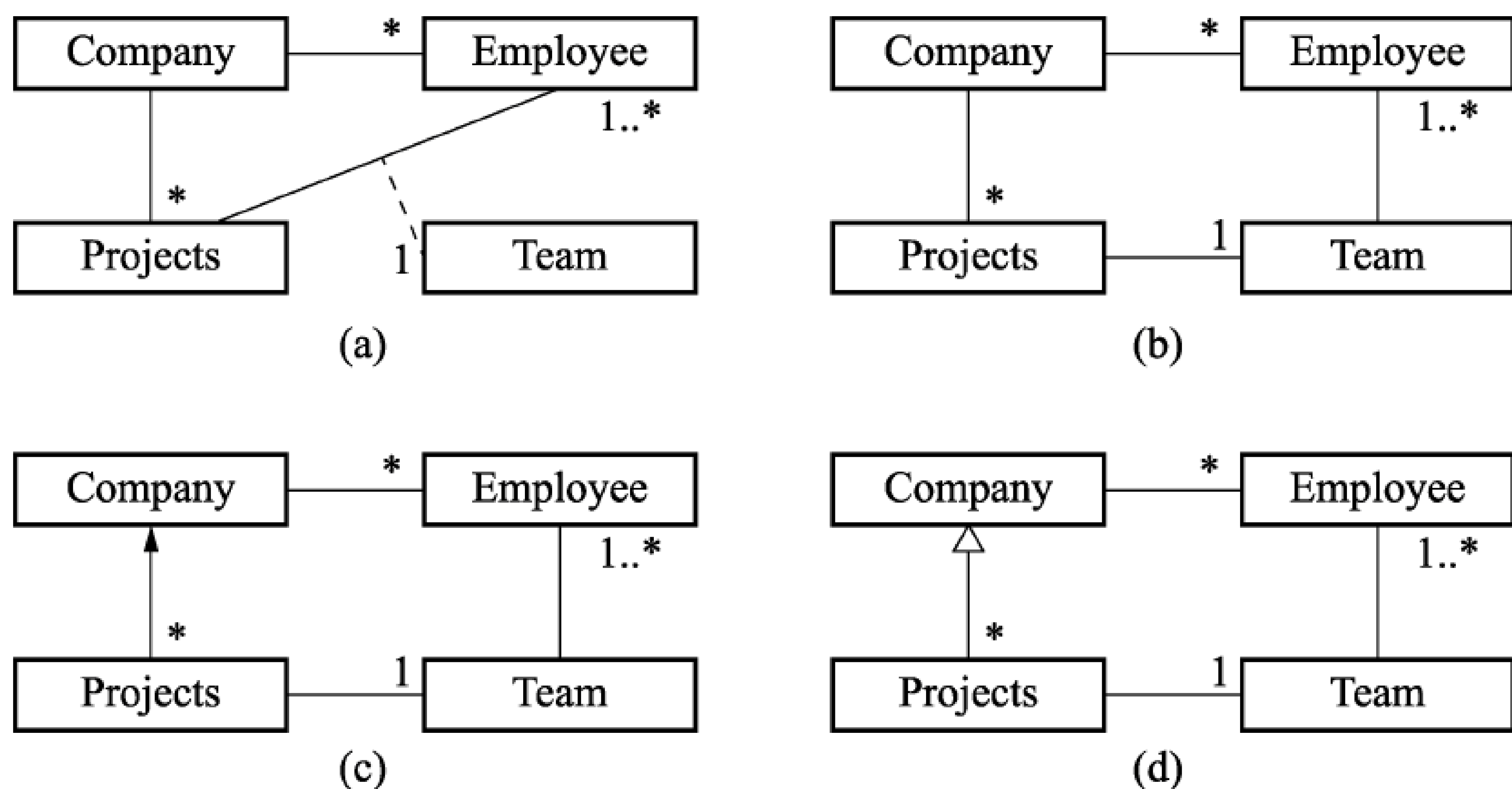
(43) A．类图　　B．部署图　　C．组件图　　D．网络图

解析：组件图(构件图)的主要目的是显示系统组件间的结构关系。构件及构件接口表示法如下图所示。



答案：C

例 5　对于场景：一个公司负责多个项目，每个项目(Project)由一个员工(Employee)团队(Team)来开发。下列 UML 概念图中，(41)最适合描述这一场景。(2012 年下半年试题 41)



(41) A．图(a)　　B．图(b)　　C．图(c)　　D．图(d)

解析：在 UML 图中，关联表示对象之间的连接，在关联上可以标注重复度和角色。选项 C 中 Company 和 Projects 之间的关系和重复度不正确；选项 D 中 Company 和 Projects 之

间不存在泛化关系；只有 B 选项最适合描述题目描述的场景。

答案：B

例 6　UML 中接口可用于<u>(42)</u>。(2012 年下半年试题 42)

(42) A．提供构造型(stereotype)名称为《terface》的具体类

B．Java 和 C++程序设计中，而 C#程序设计中不支持

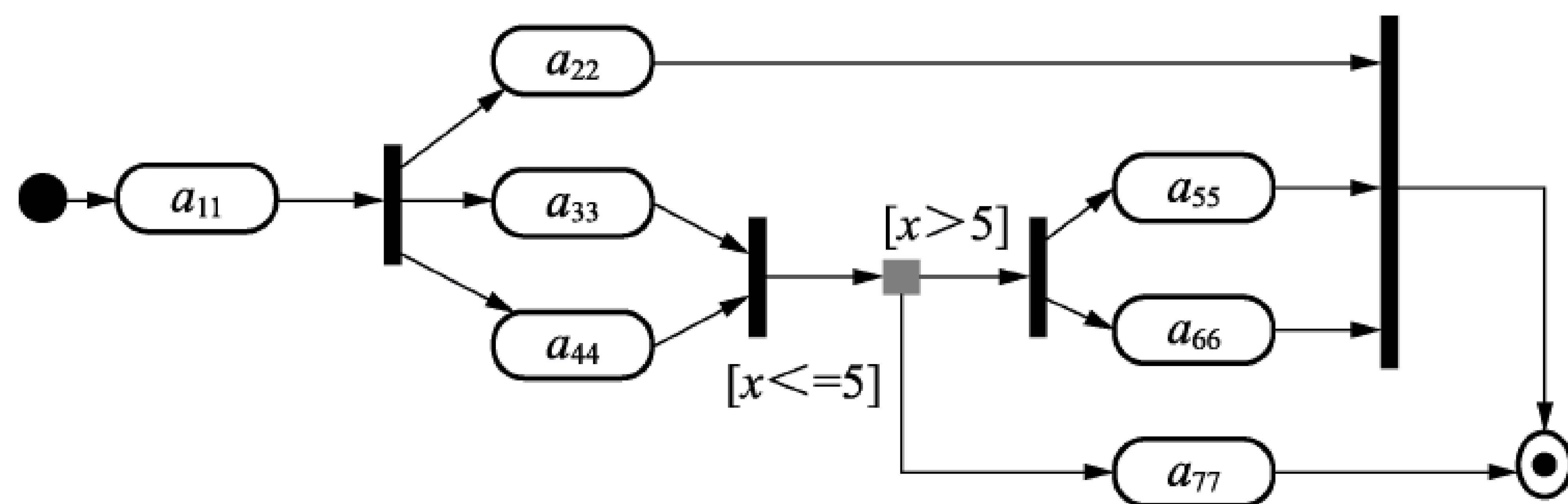
C．定义可以在多个类中重用的可执行逻辑

D．声明对象类所需要的服务

解析：接口(Interface)是描述类的部分行为的一组操作，也是一个类提供给另一个类的一组操作。和类的不同之处在于，接口只是一组操作，没有属性。要将类和接口区分开，一种办法是使用构造型<<interface>>，把它放在矩形框中接口的名字之上；另一种办法是接口的名字以大写字母“I”开头。接口可用于 Java 和 C#程序设计中，而在 C++程序设计中不支持。

答案：D

例 7　下列活动图中可以同时执行的活动是<u>(43)</u>。(2012 年下半年试题 43)



(43) A．a_{44} 和 a_{66}　　B．a_{22}、a_{33} 和 a_{44}　　C．a_{11} 和 a_{77}　　D．a_{66} 和 a_{77}

解析：活动图的主要要素包括初始节点、活动节点、活动终点、转换、分支与监护条件、分岔与汇合等。在该活动图中，a_{11}～a_{77} 为活动节点，a_{11} 后的垂直粗线段为分岔线，又称为同步线，其后的多个活动可以并发执行，因此 a_{22}、a_{33} 和 a_{44} 可以同时执行。图中的菱形条件为分支条件，用来表示满足某条件时执行某个活动。

答案：B

例 8　面向对象技术中，组合关系表示<u>(37)</u>。(2012 年上半年试题 37)

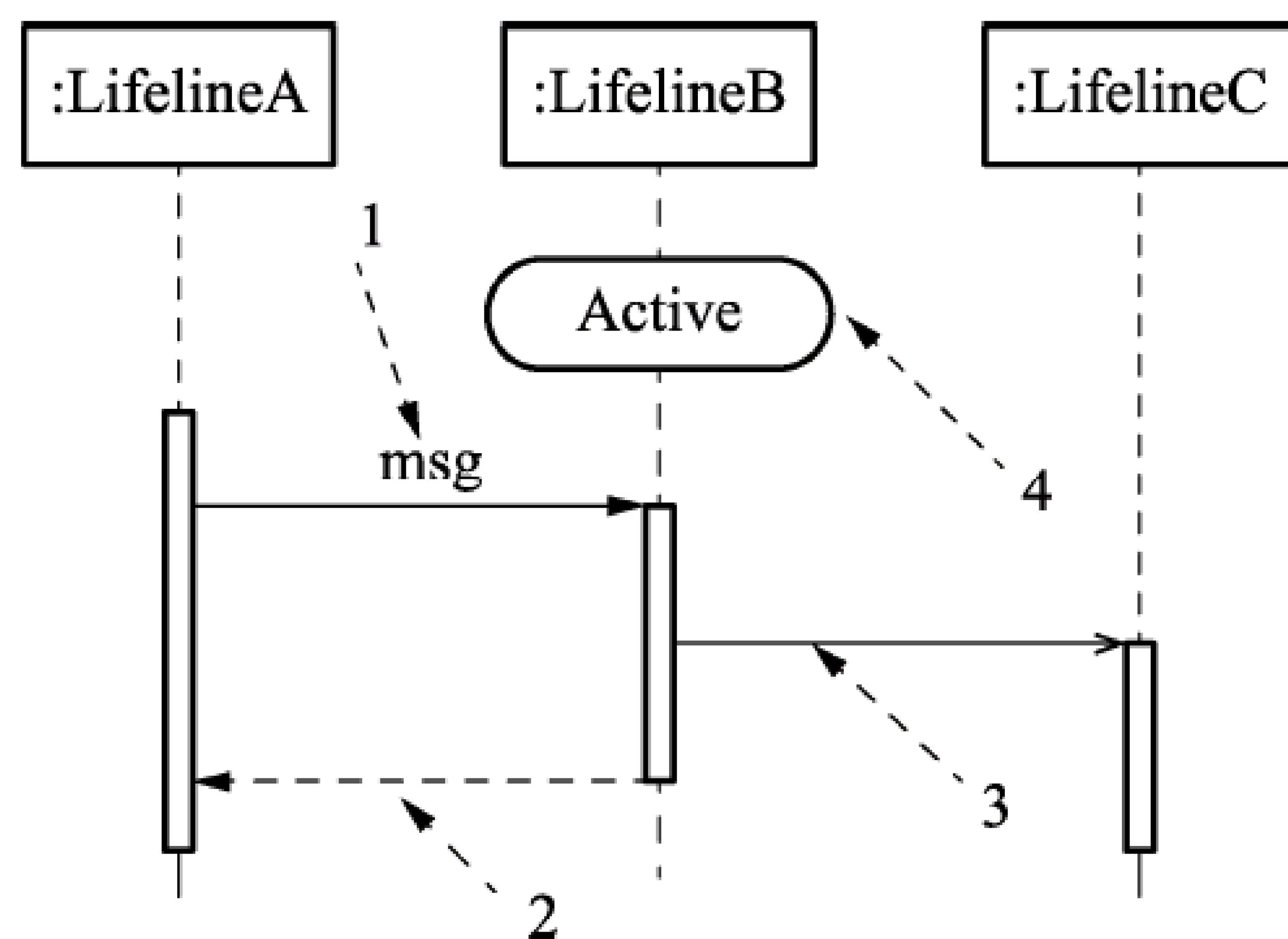
(37) A. 包与其中模型元素的关系　　B. 用例之间的一种关系

C. 类与其对象的关系　　D. 整体与其部分之间的一种关系

解析：组合关系是关联关系的一种特例，体现的是一种 contains-a 的关系，这种关系比聚合更强，也称为强聚合；它同样体现整体与部分间的关系，但此时整体与部分是不可分的，整体的生命周期结束也就意味着部分的生命周期结束，如人和大脑。组合跟聚合几乎相同，唯一的区别就是“部分”不能脱离“整体”而单独存在，也就是说，“部分”的生命期不能比“整体”还要长。

答案：D

例 9　UML 序列图是一种交互图，描述了系统中对象之间传递消息的时间序列。其中，异步消息与同步消息不同，<u>(40)</u>。下图中<u>(41)</u>表示一条同步消息，<u>(42)</u>表示一条异步消息，<u>(43)</u>表示一条返回消息。(2012 年上半年试题 40～43)



(40) A. 异步消息并不引起调用者终止执行而等待控制权的返回
B. 异步消息和阻塞调用有相同的效果
C. 异步消息是同步消息的响应
D. 异步消息和同步消息一样等待返回消息

(41) A. 1　　B. 2　　C. 3　　D. 4

(42) A. 1　　B. 2　　C. 3　　D. 4

(43) A. 1　　B. 2　　C. 3　　D. 4

解析：序列图将交互关系表示为一个二维图。纵向是时间轴，时间沿竖线向下延伸。横向轴代表了在协作中各独立对象的类元角色。类元角色用生命线表示。当对象存在时，角色用一条虚线表示，当对象的过程处于激活状态时，生命线是一个双道线。消息用从一个对象的生命线到另一个对象生命线的箭头表示。箭头以时间顺序在图中从上到下排列。对于同步消息，发送人在它继续之前，将等待同步消息响应。而对于异步消息，在发送方继续之前，无须等待响应的消息。

为了可读性，序列图的第一个消息总是从顶端开始，并且一般位于图的左边。然后继发的消息加入图中，稍微比前面的消息低些。为了显示一个对象(如生命线)传递一个消息给另一个对象，画一条线指向接收对象，包括一个实心箭头(如果是一个同步调用操作)或一个棍形箭头(如果是一个异步信号)。消息/方法名字放置在带箭头的线上面。正在被传递给接收对象的消息，表示接收对象的类实现的一个操作/方法。返回消息是可选择的。一个返回消息画作一个带开放箭头的虚线，向后指向来源的生命线，在这条虚线上面，放置操作的返回值。

答案：(40) A　(41) A　(42) C　(43) B

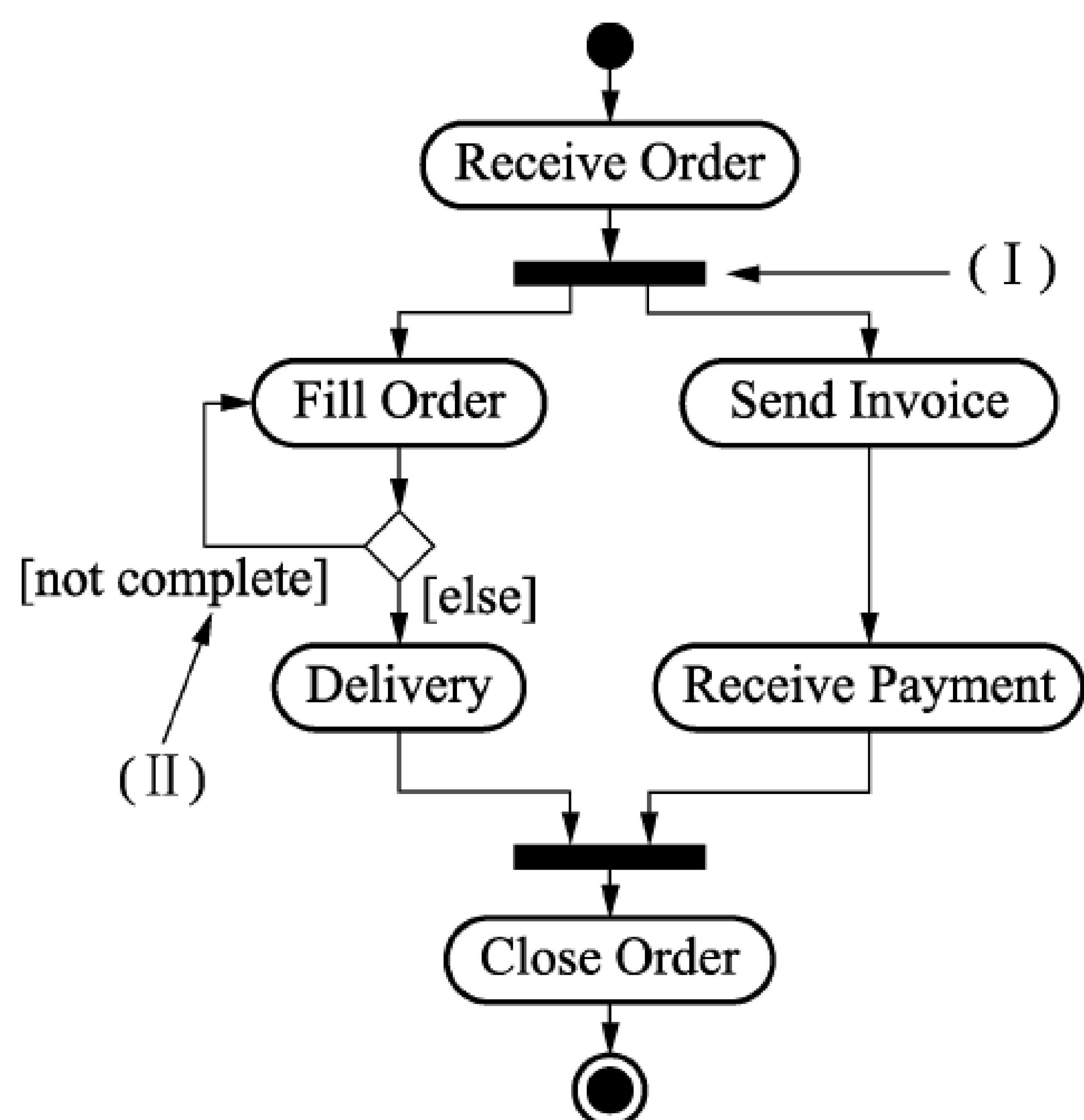
例 10　采用UML进行面向对象开发时，部署图通常在__(45)__阶段使用。(2011年下半年试题45)

(45) A. 需求分析　　B. 架构设计　　C. 实现　　D. 实施

解析：部署图展现了运行处理节点以及其中构件的配置，给出了体系结构的静态实施视图。

答案：D

例 11　如下图所示的UML图是__(41)__，图中(Ⅰ)表示__(42)__，(Ⅱ)表示__(43)__。(2016年上半年试题41、42、43)



(41) A．序列图　　B．状态图　　C．通信图　　D．活动图
(42) A．合并分叉　　B．分支　　C．合并汇合　　D．流
(43) A．分支条件　　B．监护表达式　　C．动作名　　D．流名称

解析：本题所涉及的图为活动图，该图容易与状态图混淆，对于初学者，可以把握一个原则来判断，即：状态图中每个节点对应的是状态，而状态与状态之间的变迁涉及事件触发，所以在状态图中，每条箭线上都会有事件，而活动图则不一定有。

图中 I 所代表的是同步条，也称分支，代表从此处开始有多个并行分支。而 II 所对应的是分支条件。

答案：(41) D　(42) B　(43) A

例 12　UML 中关联的多重度是指__(42)__。(2011 年上半年试题 42)

(42) A．一个类中被另一个类调用的次数
B．一个类的某个方法被另一个类调用的次数
C．一个类的实例能否与另一个类的多少个实例相关联
D．两个类所具有的相同的方法和属性

解析：关联是一种结构关系，它描述了一组链，链是对象之间的连接。关联的多重度是指一个类的实例能与另一个类的多少个实例相关联。

答案：C

例 13　__(43)__是一种很强的“拥有”关系，“部分”和“整体”的生命周期通常一样。整体对象完全支配其组成部分，包括它们的创建和销毁等；__(44)__同样表示“拥有”关系，但有时候“部分”对象可以在不同的“整体”对象之间共享，并且“部分”对象的生命周期也可以与“整体”对象不同，甚至“部分”对象可以脱离“整体”对象而单独存在。上述两种关系都是__(45)__关系的特殊种类。(2010 年下半年试题 43～45)

(43) A．聚合　　B．组合　　C．继承　　D．关联
(44) A．聚合　　B．组合　　C．继承　　D．关联
(45) A．聚合　　B．组合　　C．继承　　D．关联

解析：本题考查 UML 中关联关系。

关联关系连接元素和链接实例，它用连接两个模型元素的实线表示，在关联的两端可

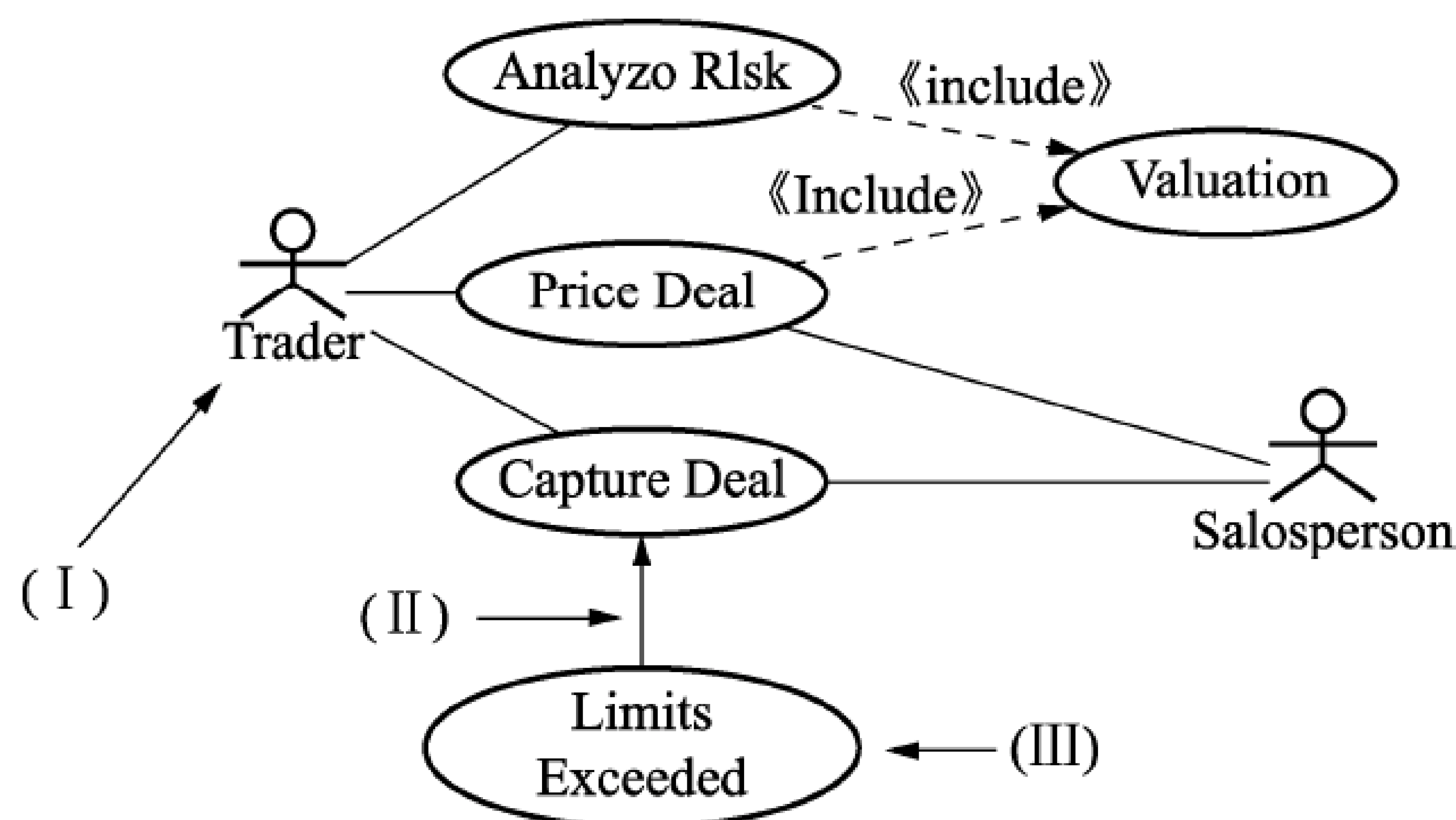
以标注关联双方的角色和多重性标记。

聚合关系是一种特殊类型的关联关系。它描述元素之间部分和整体的关系，即一个表示整体的模型元素可能由几个表示部分的模型元素聚合而成。

组合也是关联关系的一种特例，这种关系比聚合更强，也称为强聚合；它同样体现整体与部分间的关系，但此时整体与部分是不可分的，整体的生命周期结束也就意味着部分的生命周期结束。

答案：(43) B (44) A (45) D

例 14 在下图所示的 UML 图中，(I)是__(40)__，(II)是__(41)__，(III)是__(42)__。(2015 年下半年试题 40、41、42)

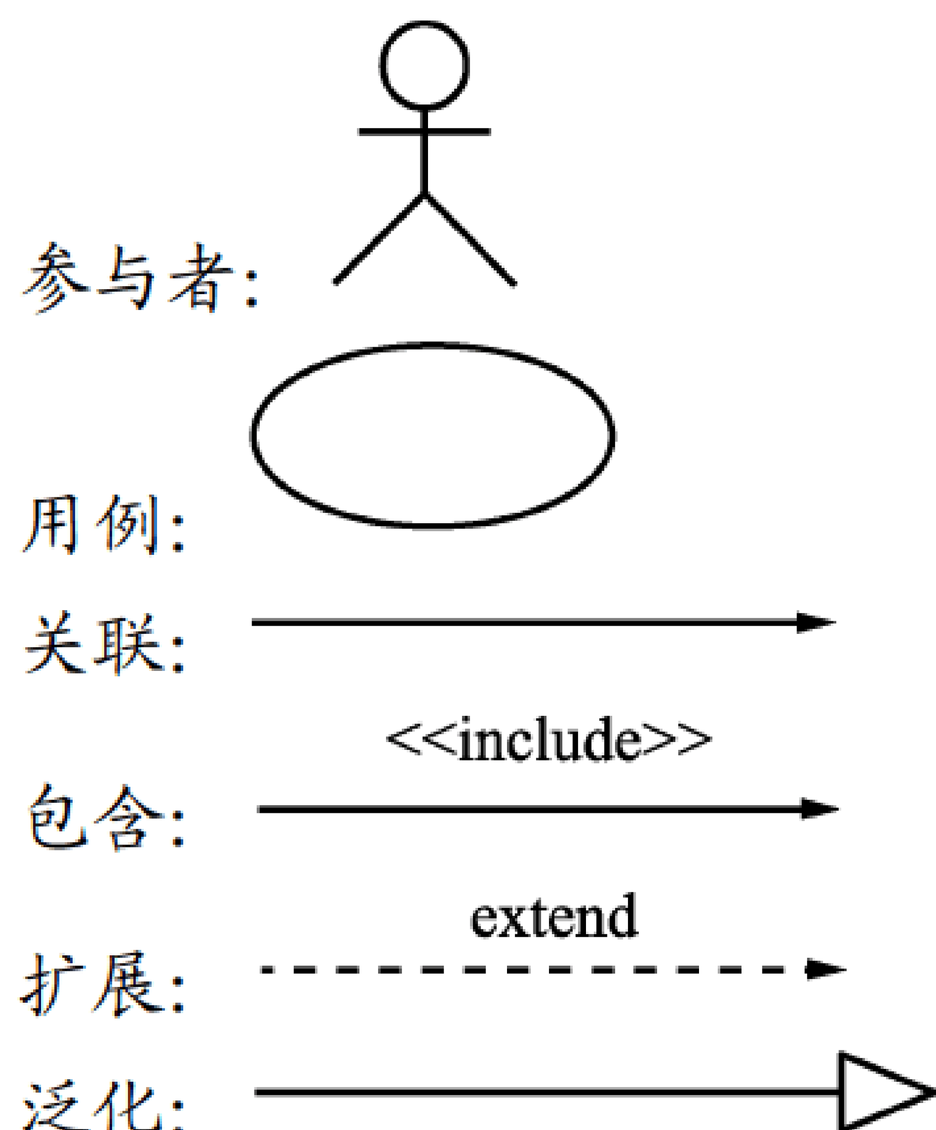


(40) A. 参与者 B. 用例 C. 泛化关系 D. 包含关系

(41) A. 参与者 B. 用例 C. 泛化关系 D. 包含关系

(42) A. 参与者 B. 用例 C. 泛化关系 D. 包含关系

解析：用例图展现了一组用例、参与者以及它们之间的关系。用例图中：



答案：(40)A (41)C (42)B

例 15 UML 图中，对新开发系统的需求进行建模，规划开发什么功能或测试用例，采用__(42)__最适合。而展示交付系统的软件组件和硬件之间关系图是__(43)__。(2015 年下半年试题 42、43)

(42) A. 类图 B. 对象图 C. 用例图 D. 交互图

(43) A. 类图 B. 部署图 C. 组件图 D. 网络图

解析：用例图用于对系统的静态用例视图进行建模，可采用两种方式：一是对系统的语境建模，包括围绕整个系统画一条线，并声明有哪些参与者位于系统之外并与系统进行交互；二是对系统的需求建模，包括说明这个系统应该做什么，而不考虑系统应该怎么做。

部署图用来显示系统中软件和硬件的物理架构。从部署图中，可以了解到软件和硬件组件之间的物理关系以及处理节点的组件分布情况。

答案：(42) C　　(43) B

例 16　UML 中有 4 种关系，即依赖、关联、泛化和实现。(40) 是一种结构关系，描述了一组链，链是对象之间的连接；(41) 是一种特殊 / 一般关系，使子元素共享其父元素的结构和行为。(2015 年下半年试题 40、41)

(40) A．依赖　　B．关联　　C．泛化　　D．实现

(41) A．依赖　　B．关联　　C．泛化　　D．实现

解析：UML 用关系把事物结合在一起，主要有下列 4 种关系。

(1) 依赖(Dependency)。依赖是两个事物之间的语义关系，其中一个事物发生变化会影响另一个事物的语义。

(2) 关联(Association)。关联描述一组对象之间连接的结构关系，对象之间的连接为链。

(3) 泛化(Generalization)。泛化是一般化和特殊化的关系，特殊元素(子元素)的对象可替换一般元素(父元素)的对象。用这种方法，子元素共享了父元素的结构和行为。

(4) 实现(Realization)。实现是类之间的语义关系，其中的一个类指定了由另一个类保证执行的契约。

答案：(40) B　　(41) C

7.4.3　同步练习

1. 在面向对象系统中，用_____关系表示一个较大的“整体”类包含一个或多个较小的“部分”类。

A．泛化　　B．聚合　　C．概化　　D．合成

2. 在 UML 的各种视图中，(1) 显示外部参与者观察到的系统功能；(2) 从系统的静态结构和动态行为角度显示系统内部如何实现系统的功能；(3) 显示的是源代码以及实际执行代码的组织结构。

(1)～(3) A．用例视图　　B．进程视图

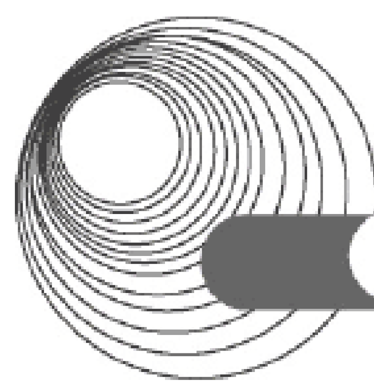
C．实现视图　　D．逻辑视图

3. 采用 UML 进行软件设计时，可用_____关系表示两类事物之间存在的特殊/一般关系，用聚集关系表示事物之间存在的整体/部分关系。

A．依赖　　B．聚集　　C．泛化　　D．实现

7.4.4　同步练习参考答案

1. B



2．(1) A　(2) D　(3) C

3．C

7.5 设计模式

7.5.1 考点辅导

7.5.1.1 设计模式的要素

设计模式一般有以下 4 个要素。

(1) 模式名称(Pattern Name)。一个助记名，它用一两个词来描述模式的问题、解决方案和效果。命名一个新的模式增加了设计词汇。设计模式允许在较高的抽象层次上进行设计。基于一个模式词汇表，就可以讨论模式并在编写文档时使用它们。模式名可以帮助人们思考，便于人们与其他人交流设计思想及设计结果。

(2) 问题(Problem)。描述了应该在何时使用模式。它解释了设计问题和问题存在的前因后果，可能描述了特定的设计问题，如怎样用对象表示算法等；也可能描述了导致不灵活设计的类或对象结构。

(3) 解决方案(Solution)。描述了设计的组成成分、它们之间的相互关系及各自的职责和协作方式。因为模式就像一个模板，可应用于多种不同场合，所以解决方案并不描述一个特定而具体的设计或实现，而是提供设计问题的抽象描述和怎样用一个具有一般意义的元素组合(类或对象组合)来解决这个问题。

(4) 效果(Consequences)。描述了模式应用的效果及使用模式应权衡的问题。尽管描述设计决策时并不总提到模式效果，但它们对于评价设计选择和理解使用模式的代价及好处具有重要意义。

7.5.1.2 创建型设计模式

创建型模式抽象了实例化过程。它们可以帮助一个系统独立于如何创建、组合和表示它的那些对象。一个类创建型模式使用继承改变被实例化的类，而一个对象创建型模式将实例化委托给另一个对象。

创建型模式中有两个不断出现的主旋律：第一，它们都将关于该系统使用哪些具体的类的信息封装起来；第二，它们隐藏了这些类的实例是如何被创建和放在一起的。整个系统关于这些对象所知道的是由抽象类所定义的接口。因此，创建型模式在什么地方被创建、谁创建它、它是怎样被创建的以及何时创建这些方面给予了很大的灵活性。它们允许用结构和功能差别很大的“产品”对象配置一个系统。配置可以是静态的(即在编译时指定)，也可以是动态的(在运行时)。

7.5.1.3 结构性设计模式

结构性模式涉及如何组合类和对象以获得更大的结构。结构性模式采用继承机制来组合接口或实现。结构性对象模式不是对接口和实现进行组合，而是描述了如何对一些对象

进行组合，从而实现新功能的一些方法。

Composite 模式是结构性对象模式的一个实例。它描述了如何构造一个类层次式结构，这一结构由两种类型的对象所对应的类构成。

Flyweight 模式为共享对象定义了一个结构。至少有两个原因要求对象共享，即效率和一致性。Flyweight 模式的对象共享机制主要强调对象的空间效率。使用很多对象的应用必须考虑每一个对象的开销。

Facade 模式描述了如何用单个对象表示整个子系统。模式中的 Facade 用来表示一组对象，Facade 的职责是将消息转发给它所表示的对象。

Bridge 模式将对象的抽象和其实现分离，从而可以独立地改变它们。

Decorator 模式描述了如何动态地为对象添加职责。这一模式采用递归方式组合对象，允许添加任意多的对象职责。

7.5.1.4　行为设计模式

行为模式涉及算法和对象间职责的分配。行为模式不仅描述对象或类的模式，还描述它们之间的通信模式。这些模式刻画了在运行时难以跟踪的复杂控制流。它们将你的注意力从控制流转移到对象间的联系方式上来。

行为类模式使用继承机制在类间分派行为，主要有 TemplateMethod 和 Interpreter 两种模式。

行为对象模式使用对象复合而不是继承。一些行为对象模式描述了一组对等的对象怎样相互协作以完成其中任一个对象都无法单独完成的任务。

Observer 模式定义并保持对象间的依赖关系。典型的 Observer 的例子就是 Smalltalk 中的模型/视图/控制器，其中一旦模型的状态发生变化，模型的所有视图都会得到通知。

其他的行为对象模式常将行为封装在一个对象中，并将请求指派给它。

7.5.2　典型例题分析

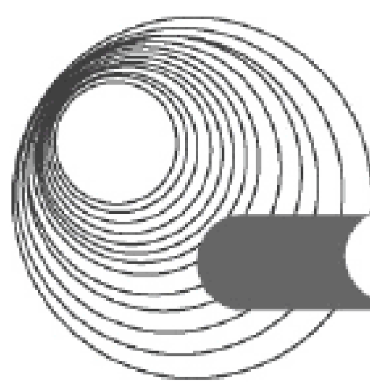
例 1　(44) 设计模式能使一个对象的状态发生改变时通知所有依赖它的监听者。(45) 设计模式限制类的实例对象只能有一个。适配器(Adapter)设计模式可以用于 (46) 。用于一个对象添加更多功能而不适用子类的是 (47) 设计模式。(2013 年上半年试题 44～47)

(44) A. 责任链(Chain of Responsibility)　　B. 命令(Command)
　　C. 抽象工厂(Abstract Factory)　　D. 观察者(Observer)

(45) A. 原型(Prototype)　　B. 工厂方法(Factory Method)
　　C. 单例(Singleton)　　D. 生成器(Builder)

(46) A. 将已有类的接口转换成和目标接口兼容
　　B. 改进系统性能
　　C. 将客户端代码数据转换成目标接口期望的合适的格式
　　D.使所有接口不兼容类可以一起工作

(47) A. 桥接(Bridge)　　B. 适配器(Adapter)
　　C. 组合(Composite)　　D. 装饰器(Decorator)



解析：观察者模式(有时又称为发布-订阅模式、模型-视图模式、源-收听者模式或从属者模式)是软件设计模式的一种。在此种模式中，一个目标物件管理所有相依于它的观察者物件，并且在它本身的状态改变时主动发出通知。这通常通过呼叫各观察者所提供的方法来实现。

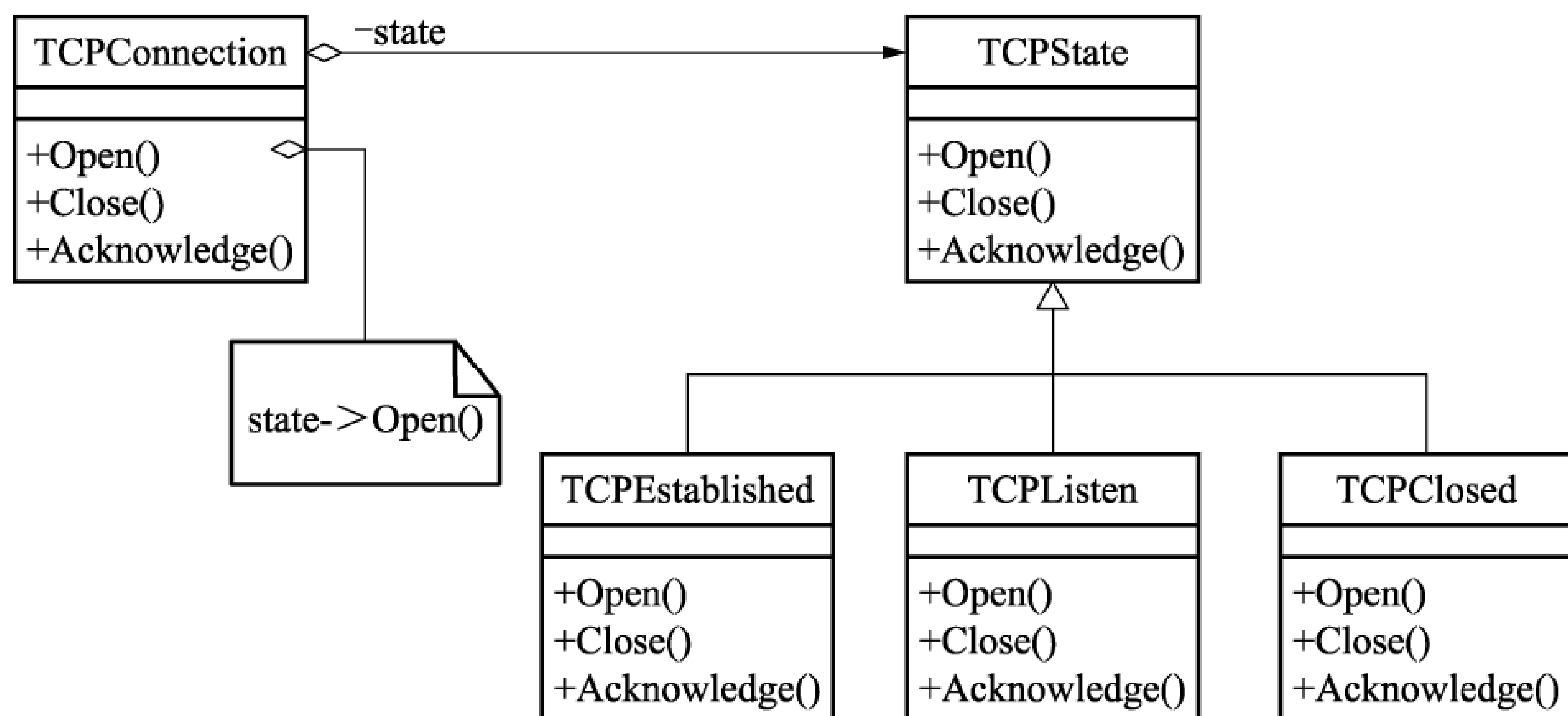
单例模式是一种常用的软件设计模式。在它的核心结构中只包含一个被称为单例类的特殊类。通过单例模式可以保证系统中一个类只有一个实例而且该实例易于外界访问，从而方便对实例个数的控制并节约系统资源。

适配器模式(有时也称包装样式或者包装)将一个类的接口适配成用户所期待的。一个适配允许通常因为接口不兼容而不能在一起工作的类工作在一起，做法是将类自己的接口包裹在一个已存在的类中。

装饰器模式就是创建一个新类为某一个类动态添加新功能或增强原有的功能。

答案：(44) D　(45) C　(46) A　(47) D

例 2　每种设计模式都有特定的意图。(44)模式使得一个对象在其内部状态改变时通过调用另一个类中的方法改变其行为，使这个对象看起来如同修改了它的类。下图是采用该模式的有关 TCP 连接的结构图实例。该模式的核心思想是引入抽象类(45)来表示 TCP 连接的状态，声明不同操作状态的公共接口，其子类实现与特定状态相关的行为。当一个(46)对象收到其他对象的请求时，它根据自身的当前状态作出不同的反应。(2012 年下半年试题 44～46)



(44) A．适配器(Adapter)　　B．命令(Command)
　　C．观察者(Observer)　　D．状态 (State)

(45) A．TCPConnection　B．state　C．TCPState　D．TCPEstablished

(46) A．TCPConnection　B．state　C．TCPState　D．TCPEstablished

解析：状态 (State)模式允许一个对象在其内部状态改变时改变它的行为，使这个对象看起来如同修改了它的类。状态模式需要对每一个系统可能取得的状态创立一个状态类的子类，当系统的状态发生变化时，系统便改变所选的子类。在上图中，TCPConnection 为情景类，定义客户应用程序感兴趣的接口，维护一个具体状态(TCPEstablished、TCPListen 和 TCPClosed)子类的实例对象。TCPState 为抽象状态类，定义一个接口用来封装与 TCPConnection 的一个特别状态(State)相关的行为。TCPEstablished、TCPListen 和 TCPClosed

为具体类，每一个具体类实现了一个 TCPConnection 的状态相关的行为。

答案：(44) D　(45) C　(46) A

例 3　欲使类 A 的所有使用者都使用 A 的同一个实例，应 (47) 。(2012 年下半年试题 47)

(47) A．将 A 标识为 final

B．将 A 标识为 abstract

C．将单例(Singleton)模式应用于 A

D．将备忘(Memoto)模式应用于 A

解析：单例(Singleton)模式用于确保某个类只有一个实例，且自行实例化，并向整个系统提供该实例。Singleton 模式为一个面向对象的应用程序提供了对象唯一的访问点，不管它实现何种功能，整个应用程序都会共享一个实例对象。

答案：C

例 4　设计模式根据目的进行分类，可以分为创建型、结构型和行为型 3 种。其中结构型模式用于处理类和对象的组合。 (44) 模式是一种结构型模式。(2012 年上半年试题 44)

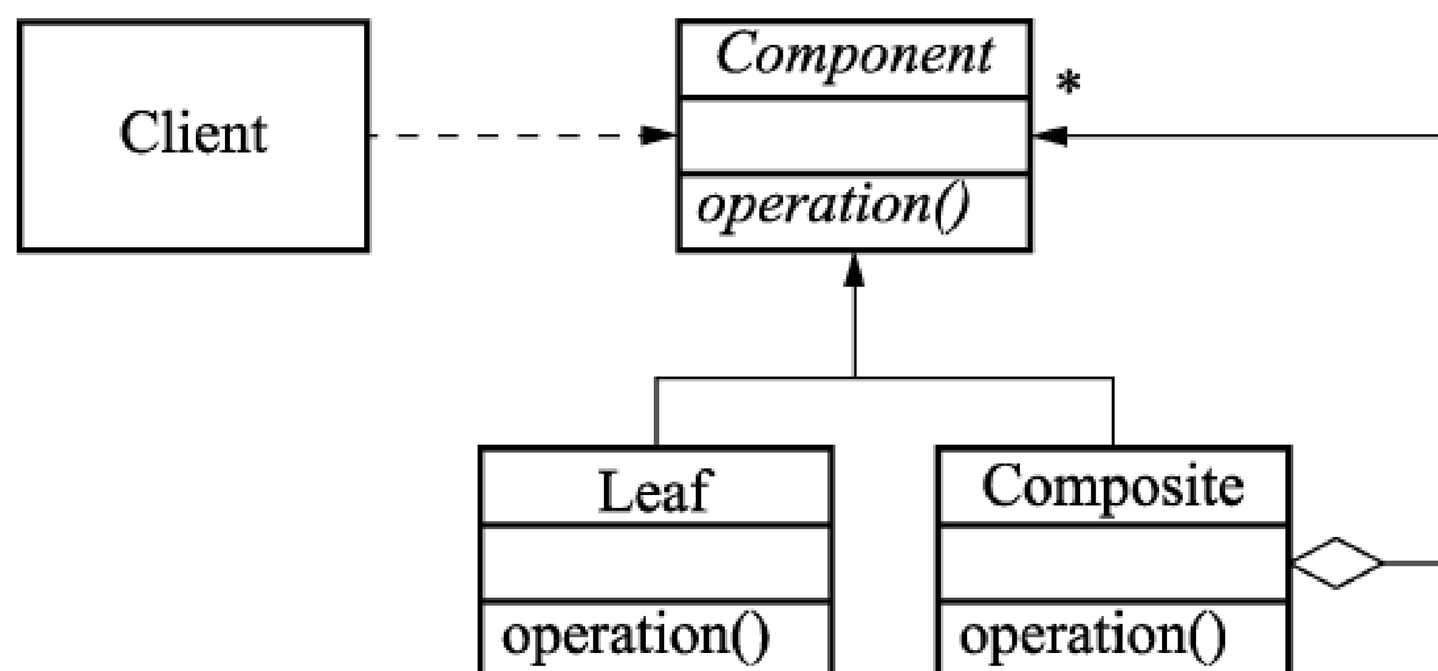
(44) A. 适配器(Adapter)　B. 命令(Command)

C. 生成器(Builder)　D. 状态(State)

解析：结构型设计模式有适配器(Adapter)模式、桥接(Bridge)模式、组合(Component)模式、代理(Proxy)模式、享元(Flyweight)模式、Facade 模式、装饰(Decorator)模式。命令(Command)和状态(State)模式是行为型模式，生成器(Builder)是创建型模式。

答案：A

例 5　下图所示为 (44) 设计模式，属于 (45) 设计模式，适用于 (46) 。(2015 年上半年试题 44、45、46)



(44) A．代理(Proxy)　B．生成器(Builder)

C．组合(Composite)　D．观察者(Observer)

(45) A．创建型　B．结构型　C．行为　D．结构型和行为

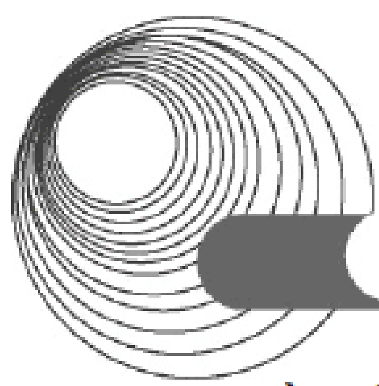
(46) A．表示对象的部分-整体层次结构时

B．当一个对象必须通知其他对象，而它又不能假定其他对象是谁时

C．当创建复杂对象的算法应该独立于该对象的组成部分及其装配方式时

D．在需要比较通用和复杂的对象指针代替简单的指针时

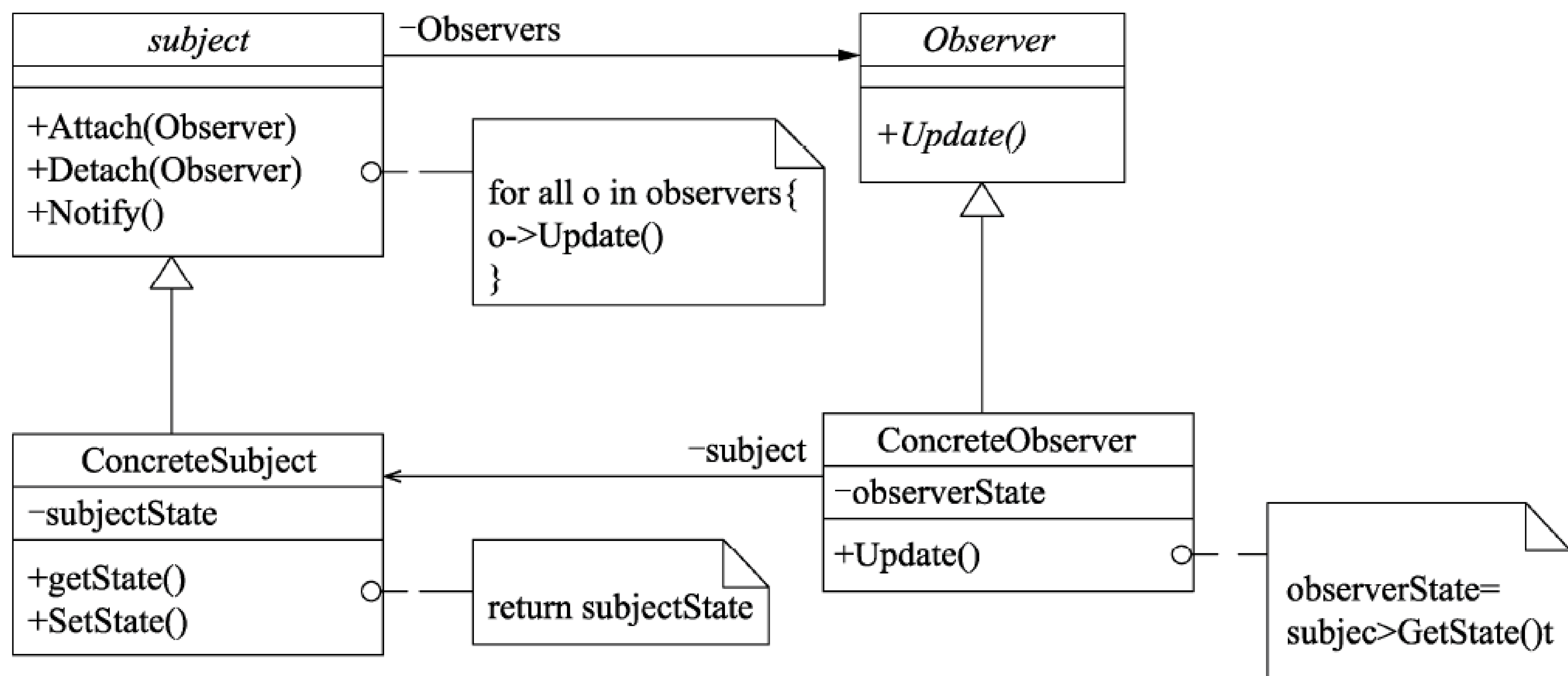
解析：结构型设计模式采用继承机制来组合接口或者实现。组合模式是一种结构型对象的一个实例，描述了如何构造一个类层次结构。当对象结构中存在“父子”关系时，可以考虑使用组合模式。组合模式将对象以树型结构组织起来，以达成“部分-整体”的层次结构，使得客户端对单个对象和组合对象的使用具有一致性。也就是说，组合模式把部分与整体的关系用树型结构表示出来，使得客户端把一个个单独的成分对象和由它们复合而



成的合成对象同等看待。

答案：(44) C　(45) B　(46) A

例 6　(40)设计模式允许一个对象在其状态改变时，通知依赖它的所有对象。该设计模式的类图如下图所示，其中，(41) 在其状态发生改变时，向它的各个观察者发出通知。(2011 年下半年试题 40、41)



(40) A. 命令(Command)　B. 责任链(Chain of Responsibility)
C. 观察者(Observer)　D. 迭代器(Iterator)

(41) A. Subject　B. ConcreteSubject
C. Observer　D. ConcreteObserver

解析：Observer 模式定义并保持对象间的依赖关系，它允许一个对象在其状态改变时，通知依赖它的所有对象。

Command 模式将请求封装在对象中，把它作为参数来传递。Iterator 抽象了访问和遍历一个集合中对象的方式。

Chain of Responsibility 是用一系列类(Classes)试图处理一个请求 request，这些类之间是一个松散的耦合，唯一共同点是在它们之间传递 request。也就是说，来了一个请求，A 类先处理，如果 A 类没有处理，就传递到 B 类处理，如果 B 类没有处理，就传递到 C 类处理，就这样像一个链条(Chain)一样传递下去。引入责任链的目的是为了解除耦合。

在观察者设计模式中，Subject 和 Observor 分别表示抽象的被观察者和观察者。通常一个观察者观察一个被观察者，而一个被观察者可以被多个观察者关注。但 Subject 的状态发生变化时，Subject 将通知所有的 Observor，告知其状态已发生变化，而 Observor 收到通知后将查询 Subject 的状态。

答案：(40) C　(41) B

例 7　欲动态地给一个对象添加职责，宜采用 (43) 模式。(2011 年下半年试题 43)

(43) A. 适配器(Adapter)　B. 桥接(Bridge)
C. 组合(Composite)　D. 装饰器(Decorator)

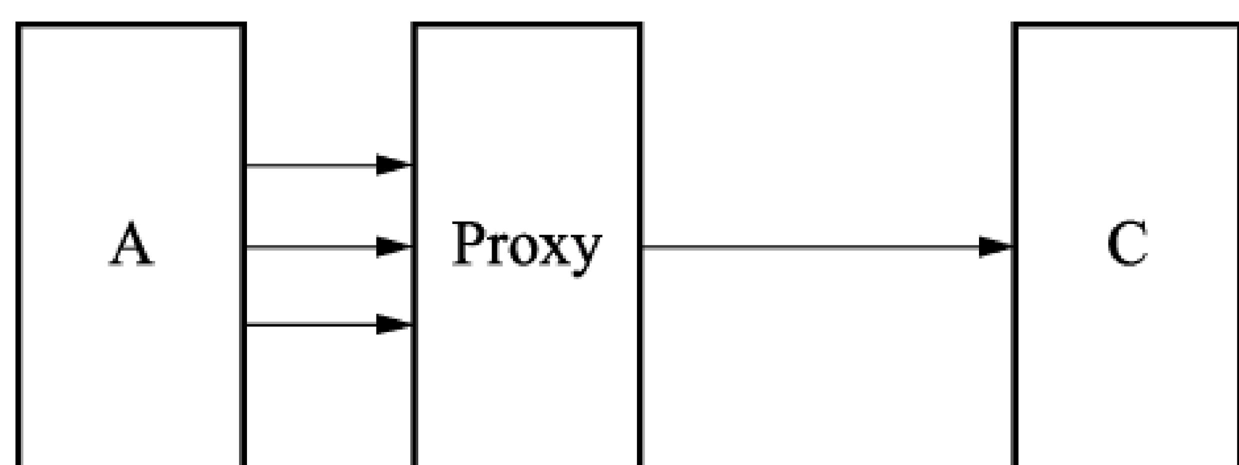
解析：适配器(Adapter)、桥接(Bridge)、组合(Composite)、装饰器(Decorator)都是结构型设计模式，用于处理类或对象的组合。适配器使得一个接口与其他接口兼容，从而给出了多个不同接口的统一抽象；桥接模式将对象的抽象和其实现分离；组合模式描述了如何构造一个类层次式结构；装饰器模式描述了如何动态地为对象添加职责。

答案： D

例 8　(44) 模式通过提供与对象相同的接口来控制对这个对象的访问。(2011 年下半年试题 44)

(44) A. 适配器(Adapter)　　B. 代理(Proxy)
　　C. 组合(Composite)　　D. 装饰器(Decorator)

解析： Proxy 代理模式是一种结构型设计模式，主要解决的问题是：在直接访问对象时带来的问题，比如说：要访问的对象在远程的机器上。在面向对象系统中，有些对象由于某些原因(如对象创建开销很大，或者某些操作需要安全控制，或者需要进程外的访问)，直接访问会给使用者或者系统结构带来很多麻烦，可以在访问此对象时加上一个对此对象的访问层，如下图所示。



答案： B

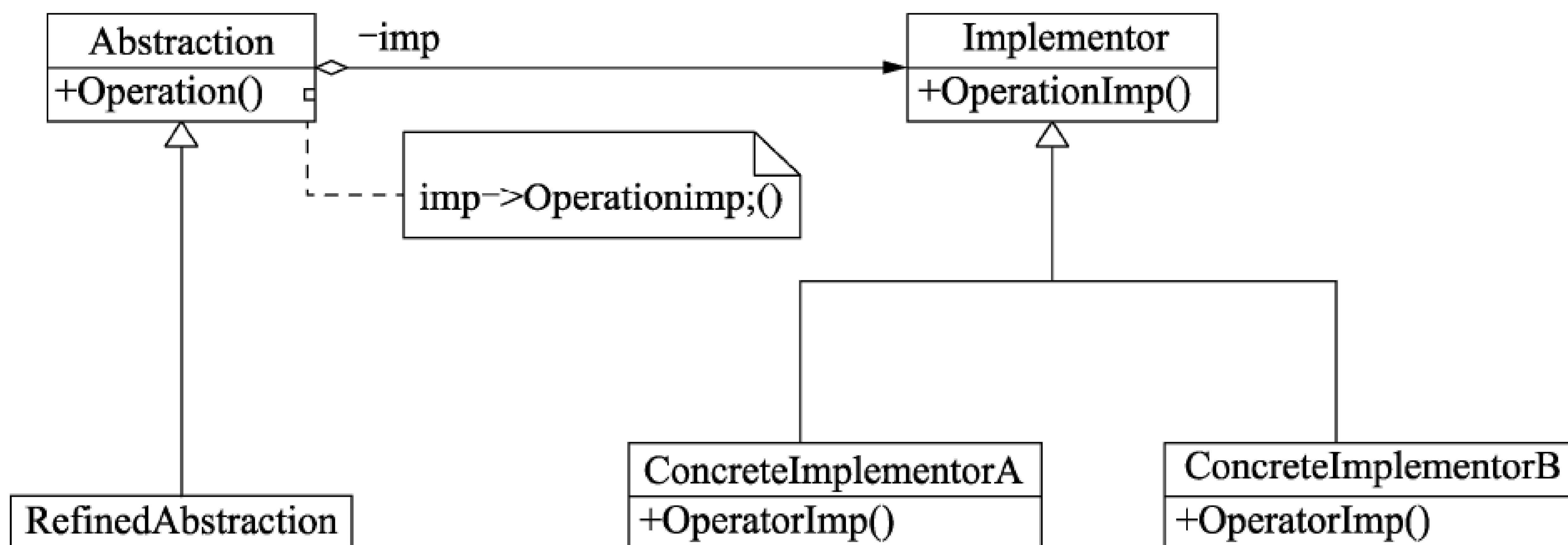
例 9　在面向对象软件开发过程中，采用设计模式 (43) 。(2011 年上半年试题 43)

(43) A. 以复用成功的设计
　　B. 以保证程序的运行速度达到最优值
　　C. 以减少设计过程创建的类的个数
　　D. 允许在非面向对象程序设计语言中使用面向对象的概念

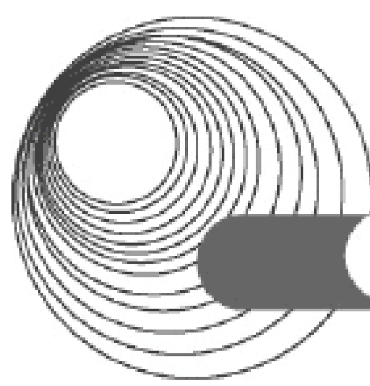
解析： 在面向对象软件开发过程中，每一个设计模式都系统地命名、解释和评价了面向对象系统中一个重要的和重复出现的设计。设计模式使人们可以更加简单、方便地复用成功的设计和体系结构。通过提供一个显式类和对象作用关系以及它们之间潜在联系的说明规范，设计模式甚至能够提高已有系统的文档管理和系统维护的有效性。简而言之，设计模式可以帮助设计者更快、更好地完成系统设计。

答案： A

例 10　设计模式 (44) 将抽象部分与其实现部分相分离，使它们都可以独立地变化。下图是该设计模式的类图，其中 (45) 用于实现部分的接口。(2011 年上半年试题 44、45)



(44) A. Bridge(桥接)　　B. Composite(组合)
　　C. Façade(外观)　　D. Singleton(单例)



(45) A．Abstraction　　B．ConcreteImplementorA
C．ConcreteImplementorB　　D．Implementor

解析： Bridge 模式将抽象部分与其实现部分相分离，使它们都可以独立地变化。Composite 模式是结构型对象模式的一个实例。它描述了如何构造一个类层次式结构，这一结构由两种类型的对象所对应的类构成。Façade 模式描述了如何用单个对象表示整个子系统。Singleton 模式保证一个类只能有一个实例。这里，前 3 种模式属于结构型设计模式，单例模式属于创建型设计模式。

在一个类图上建模继承，从子类(要继承行为的类)拉出一条闭合的、单键头(或三角形)的实线指向超类。有聚合关系的关联指出，某个类是另外某个类的一部分。在一个聚合关系中，子类实例可以比父类存在更长的时间。为了表现一个聚合关系，画一条从父类到部分类的实线，并在父类的关联末端画一个未填充菱形。本题中 Implementor 是 Abstraction 的子类，也是 Abstraction 的一部分。斜体表示 Implementor 是一个抽象类，Implementor 类使用 OperationImp()规定抽象操作，并且 ConcreteImplementorA 和 ConcreteImplementorB 两个子类分别执行它们各自版本的操作。Implementor 用于实现部分的接口。

答案： (44) A　(45) D

例 11　以下关于 Singleton(单例)模式的描述中，正确的是 (46) 。(2011 年上半年试题 46)

(46) A．它描述了只有一个方法的类的集合
B．它描述了只有一个属性的类的集合
C．它能够保证一个类的方法只能被一个唯一的类调用
D．它能够保证一个类只产生唯一的一个实例

解析： Singleton 模式要求一个类有且仅有一个实例，并且提供了一个全局的访问点。

答案： D

例 12　(47)将一个类的接口转换成客户希望的另外一个接口，使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。(2011 年上半年试题 47)

(47) A．Adapter(适配器)模式　　B．Command(命令)模式
C．Singleton(单例)模式　　D．Strategy(策略)模式

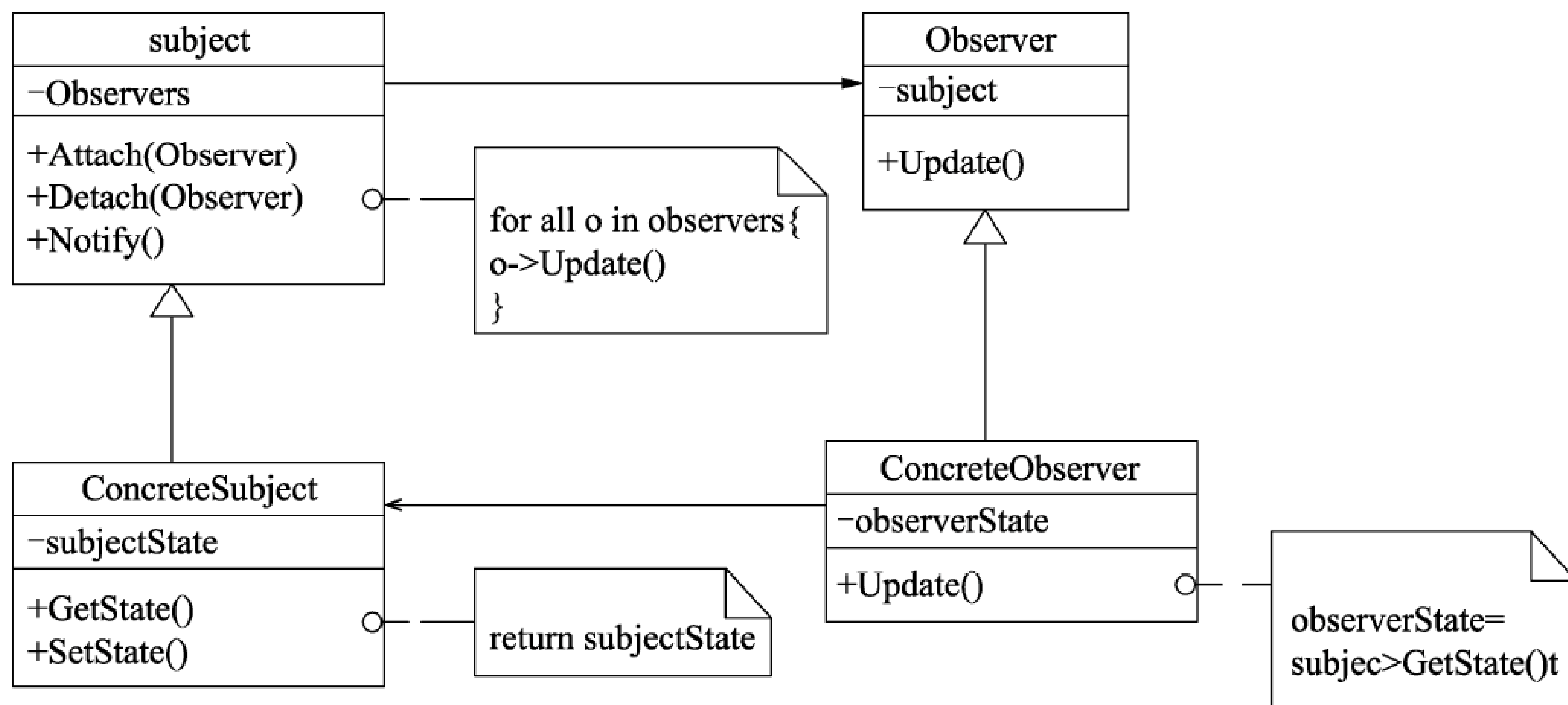
解析： 一般来说，适配器使得一个接口与其他接口兼容，从而给出了多个不同接口的统一抽象。Command 模式将请求封装在对象中，这样它就可以作为参数来传递，也可以被储存在历史列表里，或者以其他方式使用。Strategy 模式将算法封装在对象中，以方便地指定和改变一个对象所使用的算法。

答案： A

例 13　下面的 UML 类图描绘的是 (46) 设计模式。关于该设计模式的叙述错误的是 (47) 。(2010 年下半年试题 46、47)

(46) A．桥接　　B．策略　　C．抽象工厂　　D．观察者

(47) A．该设计模式中的 Observer 需要维护至少一个 Subject 对象
B．该设计模式中的 ConcreteObserver 可以绕过 Subject 及其子类的封装
C．该设计模式中一个 Subject 对象需要维护多个 Observer 对象
D．该设计模式中 Subject 需要通知 Observer 对象其自身的状态变化



解析：桥接模式将抽象部分和它的实现部分分离，使它们可以独立地变化。

策略模式定义一系列的算法，将每个算法封装到具有共同接口的独立的类中，从而使得算法可以相互替换，而不影响客户端。

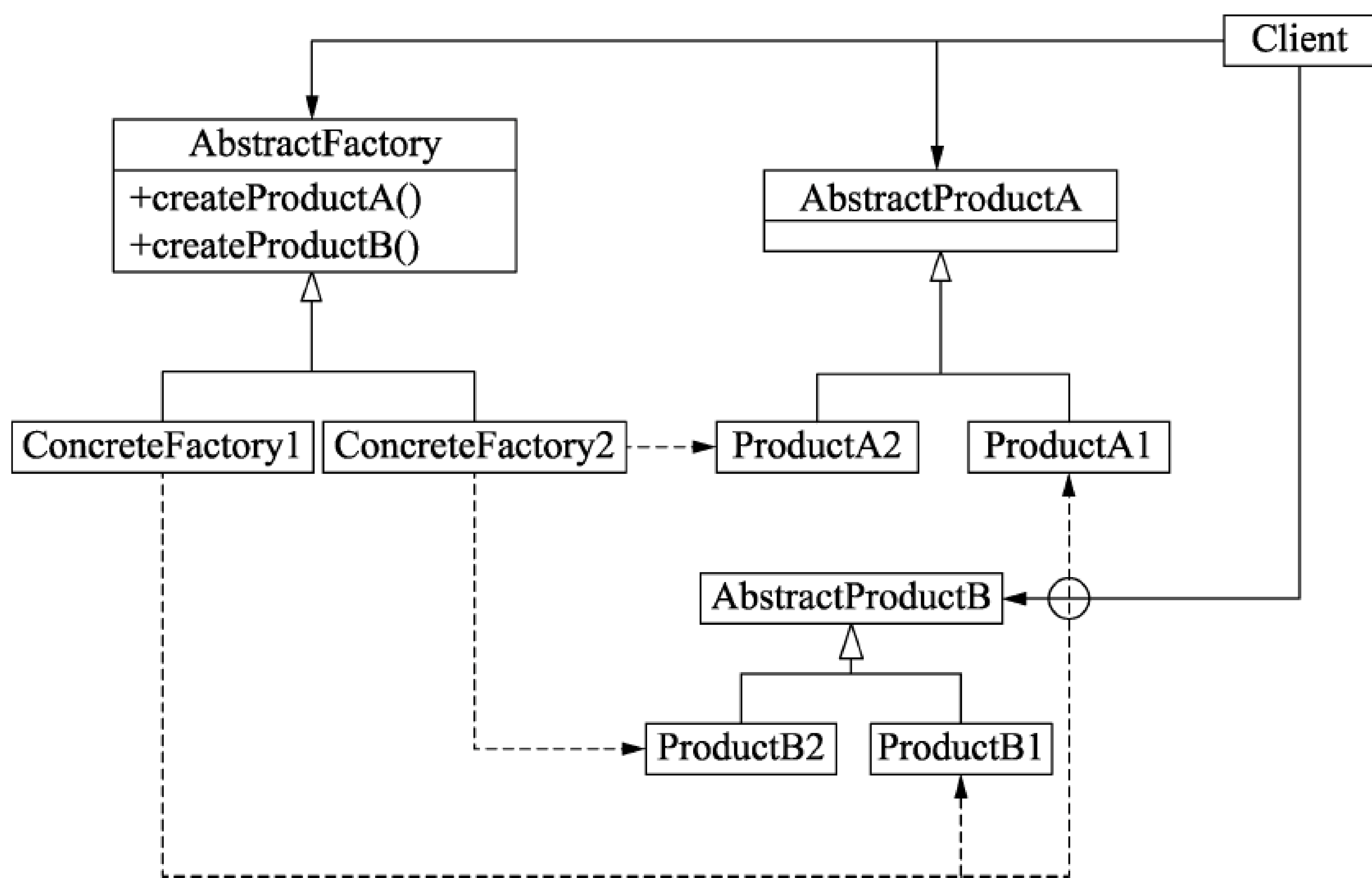
抽象工厂模式提供一个创建一系列相关或相互依赖对象的接口，而无需指定它们具体的类。

观察者模式定义对象间的一种一对多的依赖关系，以便当一个对象的状态发生变化时，所有依赖于它的对象都得到通知并自动刷新。

ConcreteObserver 类继承于 Observer 类，而 Observer 与 Subject 之间相互依赖，因此 ConcreteObserver 不可以绕过 Subject 及其子类的封装。

答案：(46) D　　(47) B

例 14　以下 UML 类图表示的是<u>(43)</u>设计模式。以下关于该设计模式的叙述错误的是<u>(44)</u>。(2010 年上半年试题 43、44)



(43) A．工厂方法　　B．策略　　C．抽象工厂　　D．观察者

(44) A．提供创建一系列相关或相互依赖的对象的接口，而无需指定这些对象所属

的具体类

B．可应用于一个系统要由多个产品系列中的一个来配置的时候

C．可应用于强调一系列相关产品对象的设计以便进行联合使用的时候

D．可应用于希望使用已经存在的类，但其接口不符合需求的时候

解析：题中的类图是抽象工厂设计模式，该设计模式的意图是提供一个创建一系列相关或相互依赖对象的接口，而无需指定它们具体的类。使用抽象工厂设计模式的常见情形是：一个系统要独立于其产品的创建、组合和表示时；一个系统要由多个产品系列中的一个来配置时；当需要强调一系列相关的产品对象的设计以便进行联合使用时；当提供一个产品类库，而只想显示它们的接口而不是实现时。对于希望使用已经存在的类，但其接口不符合需求的情形，应当考虑桥接设计模式。

答案：(43) C　　(44) D

例 15　以下关于 Singleton(单例)设计模式的叙述中，不正确的是__(44)__。(2015 年下半年试题 44)

(44) A．单例模式是创建型模式

B．单例模式保证一个类仅有一个实例

C．单例类提供一个访问唯一实例的全局访问点

D．单例类提供一个创建一系列相关或相互依赖对象的接口

解析：抽象工厂模式提供一个创建一系列相关或相互依赖对象的接口，而无须指定它们具体的类，而非单例模式。

答案：D

例 16　__(45)__设计模式能够动态地给一个对象添加一些额外的职责而无需修改此对象的结构；__(46)__设计模式定义一个用于创建对象的接口，让子类决定实例化哪一个类；欲使一个后端数据模型能够被多个前端用户界面连接，采用__(47)__模式最适合。(2015 年下半年试题 45、46、47)

(45) A．组合(Composite)　　B．外观(Facade)

C．享元(Flyweight)　　D．装饰器(Decorator)

(46) A．工厂方法(Factory Method)　　B．享元(Flyweight)

C．观察者(Observer)　　D．中介者(Mediator)

(47) A．装饰器(Decorator)　　B．享元(Flyweight)

C．观察者(Observer)　　D．中介者(Mediator)

解析：组合模式(Composite)将对象组合成树型结构以表示“整体-部分”的层次结构，使得用户对单个对象和组合对象的使用具有一致性。

外观模式(Facade)定义一个高层接口，为子系统中的一组接口提供一个一致的外观，从而简化了该子系统的使用。

享元模式(Flyweight)运用共享技术，有效地支持大量细粒度对象。

装饰模式(Decorator)动态地给一个对象添加一些额外的职责，比派生一个子类更加灵活。

工厂方法模式(Factory Method)定义一个创建对象的接口，但由子类决定需要实例化哪一个类。工厂方法使得子类实例化的过程推迟。

观察者模式(Observer)定义对象间的一种一对多的依赖关系，当一个对象的状态发生改

变时，所有依赖于它的对象都得到通知并自动更新。

中介者模式(Mediator)用一个中介对象来封装一系列的对象交互。它使各对象不需要显式地相互调用，从而达到低耦合，还可以独立地改变对象间的交互。

答案：(45) D　　(46) A　　(47) C

例 17　当不适合采用生成子类的方法对已有的类进行扩充时，可以采用(43)设计模式动态地给一个对象添加一些额外的职责；当应用程序由于使用大量的对象，造成很大的存储开销时，可以采用(44)设计模式运用共享技术来有效地支持大量细粒度的对象；当想使用一个已经存在的类，但其接口不符合需求时，可以采用(45)设计模式将该类的接口转换成希望的接口。(2009 年上半年试题 43～45)

(43)～(45) A．命令(Command)　　B．适配器(Adapter)

C．装饰(Decorate)　　D．享元(Flyweight)

解析：本题考查设计模式的相关知识，是一个重要的知识点。

命令模式把一个请求或者操作封装到一个对象中。命令模式把发出命令的责任和执行命令的责任分割开，分派给不同的对象。命令模式允许请求的一方和发送的一方独立开来，使得请求的一方不必知道接收请求的一方的接口，更不必知道请求是怎么被接受的、操作是否执行、何时被执行以及是怎么被执行的。

适配器模式把一个类的接口变换成客户端所期待的另一种接口，从而使原本因接口原因不匹配而无法一起工作的两个类能够一起工作。适配类可以根据参数返还一个合适的实例给客户端，将两个不兼容的类结合在一起使用。

享元模式以共享的方式高效地支持大量的细粒度对象。享元模式能做到共享的关键是区分内蕴状态和外蕴状态。内蕴状态存储在享元内部，不会随环境的改变而有所不同。外蕴状态是随环境的改变而改变的。外蕴状态不能影响内蕴状态，它们是相互独立的。将可以共享的状态和不可以共享的状态从常规类中区分开来，将不可以共享的状态从类里剔除出去。享元模式可大幅度降低内存中对象的数量。

装饰模式以对客户端透明的方式扩展对象的功能，是继承关系的一个替代方案，提供比继承更多的灵活性。动态地给一个对象增加功能，这些功能可以再动态地撤销。增加由一些基本功能的排列组合而产生的非常大量的功能。

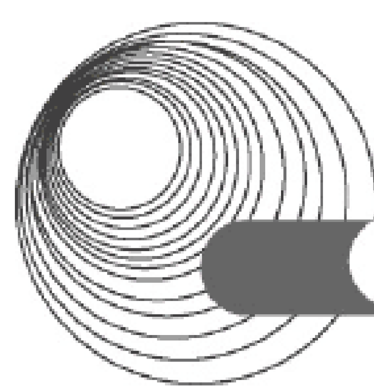
答案：(43) C　　(44) D　　(45) B

7.6　本 章 小 结

本章知识点在 2013 年的新大纲中改动不大，只是有一些描述方面的调整。

本章主要要求考生掌握面向对象的基本概念，包括面向对象的定义、类的定义、对象的概念、消息传递机制、继承机制、多态性、封装概念、面向对象程序设计以及面向对象程序设计语言中的 OOP 机制等基础知识。

面向对象所涉及的知识点很多，本章只是介绍了常用的和基础的面向对象内容。也是软件设计师的考查重点之一，每次考试大概出现 10～12 道小题，建议考生重点掌握。



7.7 达标训练题及参考答案

7.7.1 达标训练题

1．OMT 是一种对象建模技术，它定义了 3 种模型，分别是__(1)__模型、__(2)__模型和__(3)__模型，其中，__(1)__模型描述了系统中对象的表态结构，以及对象之间的联系；__(2)__模型描述系统中与时间和操作顺序有关的系统特征，表示瞬时行为上的系统的“控制”特征，通常可用__(4)__来表示；__(3)__模型描述了与值的变换有关的系统特征，通常可用__(5)__来表示。

(1) A．对象　B．功能　C．ER　D．静态
(2) A．控制　B．时序　C．动态　D．实时
(3) A．对象　B．功能　C．变换　D．计算
(4)～(5) A．类图　B．状态图　C．对象图　D．数据流图

2．一个面向对象系统的体系结构通过它的成分对象和对象间的关系确定，与传统的面向数据流的结构化开发方法相比，它具有__(1)__优点。面向对象方法有许多特征，如软件系统是由对象组成的；__(2)__；对象彼此之间仅能通过传递消息互相联系；层次结构的继承。对象模型的描述工具是__(3)__。汽车有一个发动机，汽车与发动机之间的关系是__(4)__关系。在面向对象的方法中，信息隐藏是通过对象的__(5)__来实现的。

(1) A．设计稳定　B．变换分析　C．事务分析　D．模块独立性
(2) A．开发过程基于功能分析和功能分解
B．强调需求分析的重要性
C．把对象划分成类，每个对象类都定义了一组数据和方法
D．对继承类进行调整
(3) A．状态图　B．数据流图　C．对象图　D．结构图
(4) A．一般与具体　B．整体与部分　C．分类　D．组合
(5) A．分类性　B．继承性　C．封装性　D．共享性

3．UML 是软件开发中的一个重要工具，它主要应用于__(1)__。在 UML 提供的图中，__(2)__用于对系统的静态设计视图建模，__(3)__用于对系统的静态实现视图建模。UML 中有 4 种关系，即依赖、关联、泛化和实现。其中，__(4)__是两个事务间的语义关系，即一个事务发生变化会影响另一个事务的语义。__(5)__不属于 OMT 方法所定义的模型。

(1) A．基于瀑布模型的结构化方法　B．基于需求动态定义的原型化方法
C．基于对象的面向对象的方法　D．基于数据的面向数据流的开发方法
(2)～(3) A．类图　B．对象图　C．部署图　D．构件图
(4) A．依赖　B．关联　C．泛化　D．实现
(5) A．对象模型　B．动态模型　C．功能模型　D．信息模型

7.7.2　参考答案

1．(1) A　(2) C　(3) B　(4) B　(5) D
2．(1) A　(2) A　(3) C　(4) A　(5) C
3．(1) C　(2) A　(3) D　(4) A　(5) D

第 8 章　常用算法设计方法

大纲要求：

- 排序算法、查找算法、数值计算方法、字符串处理方法、数据压缩算法、递归算法、图的相关算法。
- 算法与数据结构的关系、算法效率、算法设计、算法描述(流程图、伪代码、决策表)、算法的复杂性。

8.1　算法和算法设计的基本概念

8.1.1　考点辅导

8.1.1.1　算法

算法是对特定问题求解步骤的一种描述，它是指令的有限序列，其中每一条指令表示一个或多个操作。一个算法具有下列 5 个重要特性。

- 有穷性。一个算法必须总是在执行有穷步之后结束，且每一步都可在有穷时间内完成。
- 确定性。算法中的每一条指令必须有确切的含义，读者理解时不会产生二义性，并且在任何条件下，算法只有唯一的一条执行路径，即对于相同的输入只能得出相同的输出。
- 可行性。一个算法是可行的，即算法中描述的操作都是可以通过已经实现的基本运算执行有限次来实现的。
- 输入。一个算法有零个或多个输入，这些输入取自某个特定对象的集合。
- 输出。一个算法有一个或多个输出，这些输出是同输入有着某些特定关系的量。

8.1.1.2　算法设计

通常求解一个问题可能会有多种算法可供选择，选择的主要标准首先是算法的正确性和可靠性、简单性和易理解性；其次是算法所需要的存储空间更少和执行速度更快等。

算法设计是一件非常困难的工作，通常设计一个“好”的算法应考虑达到正确性、可读性、健壮性、效率与低存储量需求等目标。

经常采用的算法设计技术主要有迭代法、穷举搜索法、递推法、贪心法、回溯法、分治法和动态规划法等。

8.1.1.3　算法分析

算法分析是指对一个算法所需要的资源进行估算，这些资源包括内存、通信带宽、计

算机硬件和时间等，所需要的资源越多，该算法的复杂性就越高。不言而喻，对于任何给定的问题，设计出复杂性尽可能低的算法是设计算法时追求的重要目标。另外，当给定问题有很多种算法时，选择其中复杂性最低者，是选用算法时应遵循的重要准则。

在计算机资源中，最重要的是时间和空间(存储器)资源，因此复杂性分析主要包括时间复杂性和空间复杂性。

8.1.1.4　算法的表示

常用的表示算法的方法有以下几种。

(1) 自然语言。最大的优点是容易理解，缺点是容易出现二义性，并且算法通常都很冗长。

(2) 流程图。优点是直观易懂，缺点是严密性不如程序设计语言，灵活性不如自然语言。

(3) 程序设计语言。优点是能用计算机直接执行，缺点是抽象性差，使算法设计者拘泥于描述算法的具体细节，忽略了“好”算法和正确逻辑的重要性。此外，还要求算法设计者掌握程序设计语言及编程技巧。

(4) 伪代码。伪代码是介于自然语言和程序设计语言之间的方法，它采用某一程序设计语言的基本语法，操作指令可以结合自然语言来设计。

8.1.2　典型例题分析

例 1　一个算法是对某类给定问题求解过程的精确描述，算法中描述的操作都可以通过将已经实现的基本操作执行有限次来实现，这句话说明算法具有<u>(62)</u>特性。(2008 年上半年试题 62)

(62) A．有穷性　　B．可行性　　C．确定性　　D．健壮性

解析：本题考查算法的可行性。算法的可行性指的是一个算法是可行的，即算法中描述的操作都是可以通过已经实现的基本运算执行有限次来实现。

答案：B

例 2　关于算法与数据结构的关系，<u>(64)</u>是正确的。(2007 年下半年试题 64)

(64) A．算法的实现依赖于数据结构的设计
　　B．算法的效率与数据结构无关
　　C．数据结构越复杂，算法的效率越高
　　D．数据结构越简单，算法的效率越高

解析：本题考查算法和数据结构的关系。

算法是对特定问题求解步骤的一种描述，它是指令的有限序列，其中每一条指令表示一个或多个操作。算法的程序表示，也即算法要素的程序表达。一旦算法的每一项要素都用程序清楚地表达，整个算法的程序表达也就很清楚了。

数据结构是指同一数据元素类中各数据元素之间存在的关系。数据结构分别为逻辑结构、存储结构(物理结构)和数据运算。逻辑结构是对数据之间关系的描述，数据的物理(存储)结构包括数据元素的表示和关系的表示。

综上所述，算法的实现要依赖于数据结构。

答案：A

8.1.3 同步练习

下面的程序段违反了算法的_____原则。

```
void sam()
{   int n=2;
while(!odd(n))
n+=2;
printf(n);
}
```

A. 有穷性　　B. 确定性　　C. 可行性　　D. 健壮性

8.1.4 同步练习参考答案

A

8.2 算法分析基础

8.2.1 考点辅导

8.2.1.1 时间复杂性

算法的时间复杂度分析主要是分析算法的运行时间，即算法所执行的基本操作数。

即使对相同的输入规模，数据分布不相同也决定了算法执行不同的路径，因此所需要的执行时间也不相同。根据不同的输入，将算法的时间复杂度分为 3 种情况。

(1) 最佳情况。使算法执行时间最少的输入。一般情况下，不进行算法在最佳情况下的时间复杂度分析。应用最佳情况分析的一个例子是已经证明基于比较的排序算法的时间复杂度下限为 $\Omega(n\lg n)$，那么就不需要白费力气去想方设法将该类算法改进为线性时间复杂度。

(2) 最坏情况。使算法执行时间最多的输入。一般会进行算法在最坏时间复杂度的分析，因为最坏情况是在任何输入下运行时间的一个上限，它提供了一个保障，情况不会比这更糟糕。另外，对于某些算法来说，最坏情况还是相当频繁的。而且大致上看，平均情况通常与最坏情况的时间复杂度一样。

(3) 平均情况。算法的平均运行时间。一般来说，这种情况很难分析。举个简单的例子，现要排序 10 个不同的整数，输入就有 10!种不同的情况，平均情况的时间复杂度要考虑每一种输入及其该输入的概率。平均情况分析可以按以下 3 个步骤进行。

① 将所有的输入按其执行时间分类。

② 确定每类输入发生的概率。

③ 确定每类输入的执行时间。

下式给出了一般算法在平均情况下的复杂度分析，即

$$T(n)=\sum_{i=1}^{m} p_i \times t_i$$

式中，p_i为第 i 类输入发生的概率；t_i为第 i 类输入的执行时间，输入分为 m 类。

8.2.1.2　渐进符号

渐进符号有以下几种。

(1)　O 记号。给出一个函数的渐进上界。

(2)　Ω 记号。给出一个函数的渐进下界。

(3)　Θ 记号。给出一个函数的渐进上界和下界，即渐进确界。

8.2.1.3　递归式

从算法的结构上看，算法可以分为非递归形式和递归形式。非递归算法的时间复杂度分析较简单，本小节主要讨论递归算法的时间复杂度分析方法。

(1)　展开法。将递归式中等式右边的项根据递归式进行替换，称为展开。展开后的项被再次展开，如此下去，直至得到一个求和表达式及其结果。

(2)　代换法。这一名称来源于当归纳假设用较小值时，用所猜测的值代替函数的解。用代换法解递归式时需要两个步骤：猜测解的形式；用数学归纳法找出使解真正有效的常数。

(3)　递归树法。递归树法弥补了代换法猜测困难的缺点，它适于提供“好”的猜测，然后用代换法证明。在递归树中，每一个节点都代表递归函数调用集合中每一个子问题的代价。将树中每一层内的代价相加得到一个每层代价的集合，再将每层的代价相加得到递归式所有层次的总代价。当用递归式表示分治算法的时间复杂度时，递归树的方法尤其有用。

(4)　主方法。也称为主定理，给出求解以下形式的递归式的快速方法，即

$$T(n)=aT(n/b)+f(n)$$

式中，$a \geqslant 1$ 和 $b>1$ 是常数；$f(n)$是一个渐进的正函数。

8.2.2　典型例题分析

例 1　在字符串的模式匹配过程中，如果模式串的每个字符依次和主串中的一个连续的字符序列相等，则称为匹配成功。如果不能在主串中找到与模式串相同的子串，则称为匹配失败。在布鲁特—福斯模式匹配算法(朴素的或基本的模式匹配)中，若主串和模式串的长度分别为 n 和 m(且 $n \gg m$)，且恰好在主串末尾的 n 个字符处匹配成功，则在上述的模式匹配过程中，字符的比较次数最多为__(57)__。(2012 年下半年试题 57)

(57) A．$n*m$　　B．$(n-m+1)*m$　　C．$(n-m-1)*m$　　D．$(n-m)*n$

解析：在最坏情况下，每一趟不成功的匹配都是模式串的最后一个字符与主串中相应的字符不相等，则主串中新一趟的起始位置为 $i-m+2$。若从主串的第 i 个字符开始匹配时成功，则前 i 趟不成功的匹配中，每趟都比较了 m 次，总共比较了 $i*m$ 次，第 $i+1$ 趟的成功匹配也比较了 m 次。因此，在本题所述的匹配模式中，字符的比较次数最多为$(n-m+1)*m$ 次。

答案：B

例 2 在字符串的 KMP 模式匹配算法中，需要求解模式串 p 的 next 函数值，其定义如下所示。若模式串 p 为“*aaabaaa*”，则其 next 函数值为 (58) 。(2012 年上半年试题 58)

$$next[j]=\begin{cases}0 & j=1\\ \max\{k \mid 1<k<j,\ 'p_1p_2\cdots p_{k-1}'='p_{j-k+1}p_{j-k+2}\cdots p_{j-1}'\} & \\ 1 & 其他情况\end{cases}$$

(58) A. 0123123　　B. 0123210　　C. 0123432　　D. 0123456

解析：j=1 时，next[1]=0。j=2 时，不存在 k，满足 $1<k<j$，则 next[2]=1。j=3 时，k 只能取 2，等式的左边为 p_1，等式的右边为 p_2，$p_1=p_2=a$，next[3]=2。j=4 时，k 可以取 2 和 3，k 取 2 的时候，左边为 p_1，右边为 p_3，$p_1=p_3=a$；k 取 3 时，左边为 p_1p_2，右边为 p_2p_3，$p_1p_2=p_2p_3=aa$；k 取较大值 3，因此 next[4]=3。j=5 时， k 可以取 2、3、2，k 取 2 的时候，左边为 $p_1=a$，右边为 p_4=b，左右两边不等；k 取 3 的时候，左边为 $p_1p_2=aa$，右边为 $p_3p_4=ab$，左、右两边不等；k 取 4 的时候，左边为 $p_1p_2p_3=aaa$，右边为 $p_2p_3p_4=aaba$，左右两边不等，因此 next[5]=1。至此，可以判断正确的答案为 A。

答案：A

例 3 以下关于渐进符号的表示中，不正确的是 (62) 。(2012 年上半年试题 62)

(62) A. $n^2=\Theta(n^2)$　　B. $n^2=O(n^2)$　　C. $n^2=O(n)$　　D. $n^2=O(n^3)$

解析：如果存在正常数 c 和 n_0，使得当 $n\geqslant n_0$ 时，$T(n)\leqslant cf(n)$，则记为 $T(n)=O(f(n))$。T 和 f 的关系可以理解为 $f(n)$ 为 $T(n)$ 的一个上界，也可以理解为 T 至多增长得和 f 一样快。

如果存在正常数 c_1、c_2 和 n_0，使得当 $n\geqslant n_0$ 时，$c_1f(n)\leqslant T(n)\leqslant c_2f(n)$，则记为 $T(n)=\Theta(f(n))$。T 与 f 有着相同的阶数，或者两者最终以相同的阶数增长。

对于选项 A，$T(n)=f(n)=n^2$，只要 $c_2\geqslant c_1\geqslant 1$，$n_0>0$，就有 $c_1f(n)\leqslant T(n)\leqslant c_2f(n)$，因此有 $T(n)=\Theta(f(n))$，即 $n^2=\Theta(n^2)$。

对于选项 B，$T(n)=f(n)=n^2$，只要 $c\geqslant 1$，$n_0>0$，就有 $T(n)\leqslant cf(n)$，因此有 $T(n)=O(f(n))$，即 $n^2=O(n^2)$。

对于选项 D，$T(n)=n^2$，$f(n)=n^3$，只要 $c\geqslant 1$，$n_0>1$，就有 $T(n)\leqslant cf(n)$，因此有 $T(n)=O(f(n))$，即 $n^2=O(n^3)$。

对于选项 C，当 $n>1$ 时，n^2 的增长比 n 快，因此 $n^2=O(n)$的关系不成立。

答案：C

例 4 现要对 n 个实数(仅包含正实数和负实数)组成的数组 A 进行重新排列，使得其中所有的负实数都位于正实数之前。求解该问题算法的伪代码如下，则该算法的时间和空间复杂度为 (65) 。(2012 年上半年试题 65)

```
i=0;j=n-1
while i<j do
   while A[i]<0 do
      i=i+1;
   while A[j]>0 do
      j=j-1;
   if i<j do
      交换 A[i]和 A[j]
```

(65) A. $\Theta(n)$ 和 $\Theta(n)$　　B. $\Theta(1)$ 和 $\Theta(n)$
C. $\Theta(n)$ 和 $\Theta(1)$　　D. $\Theta(1)$ 和 $\Theta(1)$

解析：算法中用到了两个辅助变量 i 和 j，算法的空间复杂度为Θ(1)。在重新排列过程中，从数组的两端进行比较，从 i=0 开始判断 $A[i]$是否为负数，i 为负数的时候，i=i+1，直到 $A[i]$为正数；从 j=n−1 开始判断 $A[i]$是否为正数，如果为正数，j=j−1，直到 $A[j]$为负数。当 i<j 的时候交换 $A[i]$和 $A[j]$的值。然后继续判断 $A[i]$和 $A[j]$的值。数组 A 中的元素个数为 n，$A[i]$<0 和 $A[j]$>0 的比较次数共为 n+2，i=i+1 和 j=j−1 执行的次数最多为 n+2 次，if 语句中的 i<j 的比较以及交换 $A[i]$和 $A[j]$的操作分别最多执行 n−1 次，While 循环的条件判断至多执行 n 次。可见，算法的时间复杂度为 $\Theta(n)$。

答案：C

例 5　在 KMP 模式匹配算法中，需要求解模式串 p 的 next 函数值，其定义如下(其中，j 为模式串字符的序号)。对于模式串“*abaabaca*”，其 next 函数值序列为 (57) 。(2011 年下半年试题 57)

$$\text{next}[j]=\begin{cases}0 & j=1\\ \max\{k\mid 1<k<j,'p_1p_2\cdots p_{k-1}'='p_{j-k+1}p_{j-k+2}\cdots p_{j-1}'\} & \\ 1 & \text{其他情况}\end{cases}$$

(57) A. 01111111　　B. 01122341　　C. 01234567　　D. 01122334

解析：模式串“*abaabaca*”的 next 函数值序列如下表所示。

j	1	2	3	4	5	6	7	8
模式串	*a*	*b*	*a*	*a*	*b*	*a*	*c*	*a*
next[j]	0	1	1	2	2	3	4	1

当 j=1 时，next[1]=0；当 j=2 时，k 不存在，next[2]=1；当 j=3 时，不存在满足条件的 k，next[3]=1；当 j=4 时，p_1=p_3=a，存在 k=2，使得 $'p_1p_2\cdots p_{k-1}'='p_{j-k+1}p_{j-k+2}\cdots p_{j-1}'$，故 next[4]=2；当 j=5 时，存在 k=2，使得 $'p_1p_2\cdots p_{k-1}'='p_{j-k+1}p_{j-k+2}\cdots p_{j-1}'$，即 p_1=p_4=a，故 next[5]=2；当 j=6 时，存在 k=3，使得$'p_1p_2'='p_4p_5'$，故 next[6]=3；当 j=7 时，存在 k=4，使得$'p_1p_2\,p_3'='p_4p_5\,p_6'$，故 next[7]=4；当 j=8 时，不存在满足条件的 k 值，故 next[8]=1。

答案：B

例 6　对 n 个元素值分别为−1、0 或 1 的整型数组 A 进行升序排序的算法描述如下：统计 A 中−1、0 和 1 的个数，设分别为 n_1、n_2 和 n_3，然后将 A 中的 n_1 前个元素赋值为−1，第 n_1+1 到 n_1+ n_2 个元素赋值为 0，最后 n_3 个元素赋值为 1。该算法的时间复杂度和空间复杂度分别为 (64) 。(2011 年下半年试题 64)

(64) A. $\Theta(n)$和$\Theta(1)$　　B. $\Theta(n)$和$\Theta(n)$
C. $\Theta(n^2)$和$\Theta(1)$　　D. $\Theta(n^2)$和$\Theta(n)$

解析：本题需要用 3 个辅助变量 n_1、n_2 和 n_3 来保存数组 A 中−1、0 和 1 的个数，空间复杂度为 $\Theta(1)$。

在统计时，需要使用一次循环语句遍历数组 A。统计完成后，再使用一次循环语句遍历数组 A，并将 A 中的 n_1 前个元素赋值为−1，第 n_1+1 到 n_1+n_2 个元素赋值为 0，最后 n_3 个元素赋值为 1。数组 A 的元素个数为 n，因此算法的时间复杂度为 $\Theta(n)$。

答案：A

例 7 设算法 A 的时间复杂度可用递归式 $T(n)=\begin{cases}\Theta(1) & ,n=1\\ 7T(n/2)+n^2 & ,n>1\end{cases}$ 表示，算法 B 的时间复杂度可用递归式 $T(n)=\begin{cases}\Theta(1) & ,n=1\\ aT(n/4)+n^2 & ,n>1\end{cases}$ 表示，若要使得算法 B 渐进地快于算法 A，则 a 的最大整数为 (65) 。(2011 年下半年试题 65)

(65) A. 48　　B. 49　　C. 13　　D. 14

解析：对于算法 A，设 a=7，b=2，$f(n)$=n2，则 $\log_b a$>2，因此存在常数 e，使得 $f(n)=O(n^{\log_b a-\varepsilon})$，因此 $T(n)=\Theta(n^{\log_b a})=\Theta(n^{\log_2 7})$ 。

如果要使 B 渐进地快于算法 A，则有 $n^{\log_2 7}<n^{\log_4 a}$，得 $\log_2 7<\log_4 a$，求得 a<49，因此 a 的最大整数为 48。

在统计时，需要使用一次循环语句遍历数组 A。统计完成后，再使用一次循环语句遍历数组 A，并将 A 中的 n_1 前个元素赋值为−1，第 n_1+1 到 n_1+n_2 个元素赋值为 0，最后 n_3 个元素赋值为 1。数组 A 的元素个数为 n，因此算法的时间复杂度为 $\Theta(n)$ 。

答案：A

例 8 某算法的时间复杂度可用递归式 $T(n)=\begin{cases}\Theta(1) & ,n=1\\ 6T(n/5)+n & ,n>1\end{cases}$ 表示，若由 Θ 表示，则正确的是 (64) 。(2011 年上半年试题 64)

(64) A. $\Theta(n^{\log_5^6})$　　B. $\Theta(n^2)$　　C. $\Theta(n)$　　D. $\Theta(n^{\log_6^5})$

解析：a=6，b=5，$f(n)$=n，$\log_b a$=1.113，存在 $\varepsilon=0.113$，使得 $f(n)=O(n^{\log_b a-\varepsilon})$，因此 $T(n)=\Theta(n^{\log_b a})=\Theta(n^{\log_5 6})$ 。

答案：A

例 9 已知算法 A 的运行时间函数为 $T(n)=8T(n/2)+n^2$，其中 n 表示问题的规模，则该算法的时间复杂度为 (62) 。另已知算法 B 的运行时间函数为 $T(n)=XT(n/4)+n^2$，其中 n 表示问题的规模。对充分大的 n，若要算法 B 比算法 A 快，则 X 的最大值为 (63) 。(2014 年下半年试题 62、63)

(62) A. θ(n)　　B. θ(nlgn)　　C. $\theta(n^2)$　　D. $\theta(n^3)$

(63) A. 15　　B. 17　　C. 63　　D. 65

解析：本题需要用到主定理:

设 $a\geqslant1$ 和 $b>1$ 为常数，$f(n)$ 为函数，$T(n)$ 为定义在非负整数上的递归式，$T(n)=aT(n/b)+f(n)$，其中，n/b 指 $\lfloor n/b\rfloor$ 或 $\lceil n/b\rceil$，那么 $T(n)$ 可能有以下的渐近紧致界。

(1) 若对于某常数 $\varepsilon>0$，有 $f(n)=O(n^{\log_b a-\varepsilon})$，则 $T(n)=\Theta(n^{\log_b a})$ 。

(2) 若 $f(n)=\Theta(n^{\log_b a}\lg^k n)$，则 $T(n)=\Theta(n^{\log_b a}\lg^{k+1} n)$ 。

(3) 若对于某常数 $\varepsilon>0$，有 $f(n)=\Omega(n^{\log_b a+\varepsilon})$，且对于常数 $c<1$ 与所有足够大的 n 有 $af(b/n)\leqslant cf(n)$，则 $T(n)=\Theta(f(n))$ 。

在本题中，$a=8,b=2,\log_b a=3$，当 $\varepsilon=1$，$f(n)=n^{\log_b a-\varepsilon}=n^2$，所以 $T(n)=\Theta(n^3)$ 。

若算法 B 比算法 A 快，则算法 B 的时间复杂度要小于 $\Theta(n^3)$，这里 $a=X,b=4$，

$\log_b a = \log_4 X < 3$， $X < 64$，因此 X 的最大值为 63。

答案：(62) D　　(63) C

例 10　优先队列通常采用__(62)__数据结构实现，向优先队列中插入一个元素的时间复杂度为__(63)__。(2015 年下半年试题 62、63)

(62) A. 堆　　B. 栈　　C. 队列　　D. 线性表

(63) A. $\Theta(n)$　　B. $\Theta(1)$　　C. $\Theta(\lg n)$　　D. $\Theta(n^2)$

解析：普通的队列是一种先进先出的数据结构，元素在队列尾追加，而从队列头删除。在优先队列中，元素被赋予优先级。当访问元素时，具有最高优先级的元素最先被删除。优先队列具有最高优先级先出(largest-in，first-out)的行为特征。优先队列一般采用堆数据结构实现，排序后优先级最高的元素在堆顶。

向堆中插入一个元素时，首先将该元素写入到堆尾，即堆中最后一个元素的后面，然后经调整为一个新堆。因为在原有堆上插入一个新元素后，可能使以该元素的双亲节点为根的子树不为堆，从而使整个树不为堆，所以必须进行调整使之仍为一个堆。调整的方法如下，若新元素大于双亲节点的值，就让它们互换位置。新元素换到双亲位置后，使得以该位置为根的子树成为堆，但新元素可能还大于此位置的双亲节点的值，从而使以上一层的双亲节点为根的子树不为堆，还需要按上述方法继续调整。这样持续传递上去，直到以新位置的双亲节点为根的子树仍为一个堆或者调整到堆顶为止，此时得到的整个树又成为了一个堆。

优先队列出队时，堆顶元素会被删除。删除方法是：将最后一个元素和堆顶元素对换，然后重新调整为新堆。

向优先队列中插入一个元素的时间复杂度与调整新堆的时间复杂度一致，为 $\Theta(\lg n)$。

答案：(62) A　　(63) C

例 11　若对一个链表最常用的操作是在末尾插入节点和删除尾节点，则采用仅设尾指针的单向循环链表(不含头节点)时，__(65)__。(2010 年上半年试题 65)

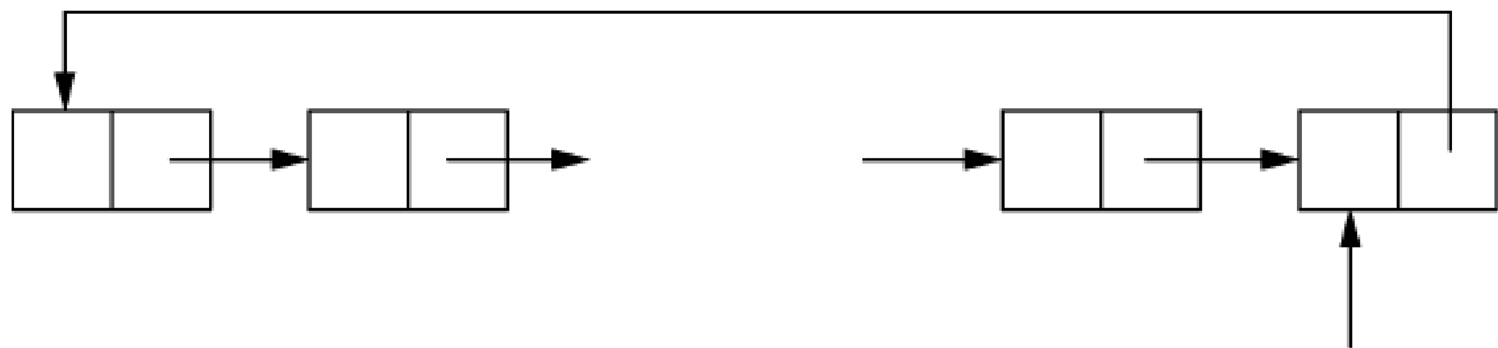
(65) A．插入和删除操作的时间复杂度都为 $O(1)$

B．插入和删除操作的时间复杂度都为 $O(n)$

C．插入操作的时间复杂度为 $O(1)$，删除操作的时间复杂度为 $O(n)$

D．插入操作的时间复杂度为 $O(n)$，删除操作的时间复杂度为 $O(1)$

解析：设尾指针的单项循环链表(不含头节点)如下图所示。



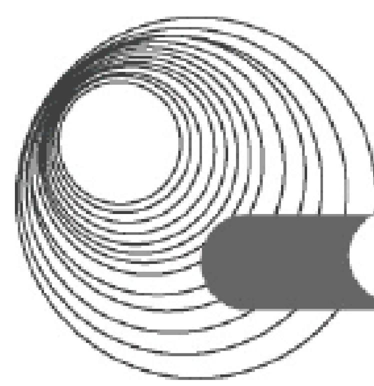
设节点的指针域为 next，新节点的指针为 s，则在尾指针所指节点后插入节点的操作如下：

```
s->next=t->next;t->next=s;t=s;
```

也就是插入操作的时间复杂度为 $O(1)$。

要删除尾指针所指节点，必须通过遍历操作找到尾节点的前驱节点，其操作序列如下：

```
if (t->next==t) free(t);
```




```
else {
   p=t->next;
   while (p->next!=t)
     p=p->next;
   p->next=t->next;
   free(t);
   t=p;
}
```

也就是说，删除操作的时间复杂度为 $O(n)$。

答案：C

例 12　某算法的时间复杂度表达式为 $T(n)=an^2+bn\lg n+cn+d$，其中，n 为问题的规模，a、b、c 和 d 为常数，用 O 表示其渐进时间复杂度为(63)。(2009 年下半年试题 63)

(63) A. $O(n^2)$　　B. $O(n)$　　C. $O(n\lg n)$　　D. $O(1)$

解析：本题考查时间复杂度的计算方法。

本题中的时间复杂度不仅与输入规模有关，还与系数 a、b、c 和 d 有关，因此对该函数做进一步的抽象，仅考虑运行时间的增长率或称为增长的量级，如忽略上式中的低阶项和高阶项的系数，因此可以得到本题的渐进时间复杂度是 $O(n^2)$。

答案：A

8.2.3　同步练习

设某算法的计算时间表示为递推关系式 $T(n)=T(n-1)+n(n>0)$ 及 $T(0)=1$，则该算法的时间复杂度为______。

A. $O(\lg n)$　　B. $O(n\lg n)$　　C. $O(n)$　　D. $O(n^2)$

8.2.4　同步练习参考答案

D

8.3　分　治　法

8.3.1　考点辅导

8.3.1.1　递归的概念

递归是指子程序(或函数)直接调用自己或通过一系列调用语句间接调用自己，是一种描述问题和解决问题的常用方法。

递归有两个基本要素：边界条件，即确定递归到何时终止，也称为递归出口；递归模式，即大问题是如何分解为小问题的，也称为递归体。

8.3.1.2　分治法的基本思想

分治法的设计思想是将一个难以直接解决的大问题分解成一些规模较小的相同问题，以便各个击破、分而治之。如果规模为 n 的问题可分解成 k 个子问题，$1<k\leq n$，这些子问题互相独立且与原问题相同。分治法产生的子问题往往是原问题的较小模式，这就为递归技术提供了方便。

一般来说，分治算法在每一层递归上都有 3 个步骤。

(1)　分解。将原问题分解成一系列子问题。

(2)　求解。递归地求解各子问题。若子问题足够小，则直接求解。

(3)　合并。将子问题的解合并成原问题的解。

8.3.2　典型例题分析

例 1　在有 n 个无序无重复元素值的数组中查找第 i 小的数的算法描述如下：任意取一个元素 r，用划分操作确定其在数组中的位置，假设元素 r 为第 k 小的数。若 i 等于 k，则返回该元素值；若 $i<k$，则在划分的前半部分递归进行划分操作找第 i 小的数；否则在划分的后半部分递归进行划分操作找第 $k-i$ 小的数。该算法是一种基于__(63)__策略的算法。(2011 年下半年试题 63)

(63) A. 分治　　B. 动态规划　　C. 贪心　　D. 回溯

解析：分治算法的基本思想是：将一个难以直接解决的大问题分解成一些规模较小的小问题以便各个击破、分而治之。分治算法的每一层都有 3 个步骤，即分解、求解和合并。本题的查找算法，不断划分数组，缩小查找范围，可见该算法是基于分治策略的算法。

答案：A

例 2　分治算法设计技术__(63)__。(2011 年上半年试题 63)

(63) A．一般由 3 个步骤组成，即问题划分、递归求解、合并解

B．一定是用递归技术来实现

C．将问题划分为 k 个规模相等的子问题

D．划分代价很小而合并代价很大

解析：分治算法的设计思想是将一个难以直接解决的大问题分解成一些规模较小的相同问题，以便各个击破、分而治之。分治算法产生的子问题往往是原问题的较小模式。一般来说，分治算法分为 3 个步骤：将原问题分解成一系列子问题；递归求解各个子问题；将子问题的解合并成原问题的解。

答案：A

例 3　现有 16 枚外形相同的硬币，其中有一枚比真币重量轻的假币，若采用分治法找出这枚假币，至少比较__(63)__次才能够找出该假币。(2009 年上半年试题 63)

(63) A．3　　B．4　　C．5　　D．6

解析：16 枚硬币分成两份(各 8 枚)，选出质量轻的那 8 枚；继续分成两份(各 4 枚)，选出质量轻的那 4 枚；继续分成两份(各 2 枚)，选出质量轻的那两枚；继续分成两份(各一枚)，选出质量轻的那一枚，即为假币。采用分治法共需比较 4 次。

答案：B

例 4　归并排序采用的算法设计方法属于(65)。(2009 年上半年试题 65)

(65) A．归纳法　　B．分治法　　C．贪心法　　D．回溯方法

解析：归并的含义是将两个或两个以上的有序表组合成一个新的有序表。假设初始序列含有 n 个记录，则可看成是 n 个有序的子序列，每个子序列的长度为 1，然后两两归并，得到 $\lceil n/2 \rceil$ 个长度为 2 或 1 的有序子序列；再两两归并，……，如此重复，直至得到一个长度为 n 的有序序列为止，这种排序方法称为 2-路归并排序。

将待排序元素分成大小大致相同的两个子集，分别对两个子集进行排序，最终将排好序的子集合并成所要求的排好序的集合。符合分治算法设计的思想。

有两种基于分治法的排序——快速排序和归并排序。

答案：B

8.3.3　同步练习

1.　给定一组长度为 n 的无序序列，将其存储在一维数组 $a[0..\ n-1]$中。现采用以下方法找出其中的最大元素和最小元素：比较 $a[0]$和 $a[n-1]$，若 $a[0]$较大，则将二者的值进行交换；再比较 $a[1]$和 $a[n-2]$，若 $a[1]$较大，则交换二者的值；然后依次比较 $a[2]$和 $a[n-3]$、$a[3]$和 $a[n-4]$、…，使得每一对元素中的较小者被交换到低下标端。重复上述方法，在数组的前 $n/2$ 个元素中查找最小元素，在后 $n/2$ 个元素中查找最大元素，从而得到整个序列的最小元素和最大元素。上述方法采用的算法设计策略是______。

A．动态规划法　　B．贪心法　　C．分治法　　D．回溯法

2.　斐波那契(Fibonacci)数列可以递归地定义为

$$F(n)=\begin{cases}1 & n=0\\ 1 & n=1\\ F(n-1)+F(n-2) & n>1\end{cases}$$

用递归算法求解 $F(5)$时需要执行 (1) 次“+”运算，该方法采用的算法策略是 (2) 。

(1) A．5　　B．6　　C．7　　D．8

(2) A. 动态规划　　B．分治　　C．回溯　　D．分支限界

3．考虑一个背包问题，共有 n=5 个物品，背包容量为 W= 10，物品的重量和价值分别为：w={2，2，6，5，4}，v={6, 3，5，4，6}，求背包问题的最大装包价值。若此为 0-1 背包问题，分析该问题具有最优子结构，定义递归式为(2016 年上半年试题 62、63、64、65)

$$c(i,j)=\begin{cases}0 & \text{若}i=0\text{或}j=0\\ c(i-1,j) & \text{若}w[i]>j\\ \max\{c(i-1,j),c(i-1,j-w(i))\} & \text{其他}\end{cases}$$

其中 $c(i,j)$表示 i 个物品、容量为 j 的 0-1 背包问题的最大装包价值，最终要求解 $c(n,W)$。

采用自底向上的动态规划方法求解，得到最大装包价值为 (62) ，算法的时间复杂度为 (63) 。

若此为部分背包问题，首先采用归并排序算法，根据物品的单位重量价值从大到小排序，然后依次将物品放入背包直至所有物品放入背包中或者背包再无容量，则得到的最大

装包价值为__(64)__，算法的时间复杂度为__(65)__。

(62) A．11　　B．14　　C．15　　D．16.67

(63) A．$\Theta(nW)$　　B．$\Theta(n\lg n)$　　C．$\Theta(n^2)$　　D．$\Theta(n\lg nW)$

(64) A．11　　B．14　　C．15　　D．16.67

(65) A．$\Theta(nW)$　　B．$\Theta(n\lg n)$　　C．$\Theta(n^2)$　　D．$\Theta(n\lg nW)$

8.3.4　同步练习参考答案

1．C　　2．(1) C　　(2) B

3．(62) C　　(63) A　　(64) D　　(65) B

8.4　动态规划法

8.4.1　考点辅导

动态规划算法与分治法类似，其基本思想也是将待求解问题分解成若干个子问题，先求解子问题，然后从这些子问题的解得到原问题的解。与分治法不同的是，适合于用动态规划法求解的问题，经分解得到的子问题往往不是独立的。

动态规划算法通常用于求解具有某种最优性质的问题。在这类问题中，可能会有许多可行解，每个解都对应于一个值，希望找到具有最优值(最大值或最小值)的那个解。当然，最优解可能会有多个，动态规划算法能找出其中的一个最优解。设计一个动态规划算法，通常可按照以下几个步骤进行。

(1) 找出最优解的性质，并刻画其结构特征。

(2) 递归地定义最优解的值。

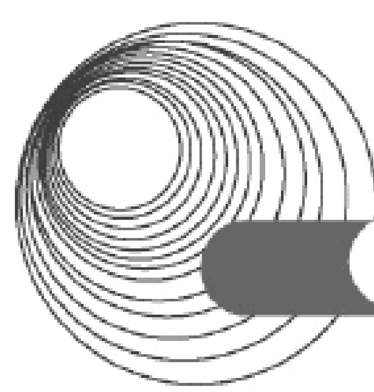
(3) 以自底向上的方式计算出最优值。

(4) 根据计算最优值时得到的信息，构造一个最优解。

对一个给定的问题，若其具有以下两个性质，则可以考虑用动态规划法来求解。

(1) 最优子结构。如果一个问题的最优解中包含其子问题的最优解，就说该问题具有最优子结构。当一个问题具有最优子结构时，表示动态规划法可能会适用，但是此时贪心策略可能也是适用的。

(2) 重叠子问题。它指用来解原问题的递归算法可反复地解同样的子问题，而不是总在产生新的子问题。即当一个递归算法不断地调用同一个问题时，就说明该问题包含重叠子问题。此时若用分治法递归求解，则每次遇到子问题都会视为新问题，会极大地降低算法的效率，而动态规划法总是充分利用重叠子问题，对每个子问题仅计算一次，把解保存在一个在需要时就可以查看的表中，而每次查表的时间为常数。



8.4.2 典型例题分析

用动态规划策略求解矩阵连乘问题 $M_1 \times M_2 \times M_3 \times M_4$，其中 $M_1(20\times5)$、$M_2(5\times35)$、$M_3(35\times4)$ 和 $M_4(4\times25)$，则最优的计算次序为 (63) 。(2010 年下半年试题 63)

(63) A．$((M_1\times M_2)\times M_3)\times M_4$　　B．$(M_1\times M_2)\times(M_3\times M_4)$

C．$(M_1\times(M_2\times M_3))\times M_4$　　D．$M_1\times(M_2\times(M_3\times M_4))$

解析：由于矩阵乘法满足结合律，故计算矩阵的连乘积可以有许多不同的计算次序，最优的计算次序是使得矩阵连乘中乘法次数最少的次序。

选项 A，乘法的次数为 20×35×5+20×4×35+20×25×4=6700

选项 B，乘法的次数为 20×35×5+35×25×4+20×25×35=24500

选项 C，乘法的次数为 5×4×35+20×4×5+20×25×4=3100

选项 D，乘法的次数为 35×25×4+5×25×35+20×25×5=10375

可见，选项 C 中的计算次序为最优的计算次序。

答案：C

8.4.3 同步练习

1. 对于求取两个长度为 n 的字符串的最长公共子序列(LCS)问题，利用 (1) 策略可以有效地避免子串最长公共子序列的重复计算，得到时间复杂度为 $O(n^2)$的正确算法。串<1, 0, 0，1, 0, 1, 0, 1>和<0, 1, 0, 1, 1, 0, 1, 1>的最长公共子序列的长度为 (2) 。

(1) A．分治　B．贪心　C．动态规划　D．分支-限界

(2) A．3　B．4　C．5　D．6

2. 利用动态规划方法求解每对节点之间的最短路径问题时，设有向图 G=<$V.E$>共有 n 个节点，节点编号为 1～n，设 C 是 G 的成本邻接矩阵，用 $D^k(I,j)$表示从 I 到 j 并且不经过编号比 K 还大的节点的最短路径的长度($D^n(I,j)$即为图 G 中节点 I 到 j 的最短路径长度)，则求解该问题的递推关系式为______。

A．$D^k(I,j)=D^{k-1}(I,j)+C(I,j)$

B．$D^k(I,j)=\min\{D^{k-1}(I,j), D^{k-1}(I,j)+C(I,j)\}$

C．$D^k(I,j)= D^{k-1}(I,j)+ D^{k-1}(k,j)$

D．$D^k(I,j)=\min\{D^{k-1}(I,j),D^{k-1}(I,k)+D^{k-1}(k,j)\}$

3. 采用动态规划策略求解问题的显著特征是满足最优性原理，其含义是______。

A．当前所作出的决策不会影响后面的决策

B．原问题的最优解包含其子问题的最优解

C．问题可以找到最优解，但利用贪心法不能找到最优解

D．每次决策必须是当前看来最优的决策才可以找到最优解

8.4.4　同步练习参考答案

1．(1) C　(2) D　2．D　3．B

8.5　贪　心　法

8.5.1　考点辅导

和动态规划法一样，贪心法也经常用于解决最优化问题。不过与动态规划法不同的是，贪心法在解决问题的策略上是仅根据当前已有的信息作出选择，而且一旦作出选择，不管将来有什么结果，这个选择都不会改变。换言之，贪心法并不是从整体最优考虑，它所作出的选择只是在某种意义上的局部最优。

用贪心法求解的问题一般具有以下两个重要的性质。

(1) 最优子结构。当一个问题的最优解包含其子问题的最优解时，称此问题具有最优子结构。问题的最优子结构是该问题可以采用动态规划法或者贪心法求解的关键性质。

(2) 贪心选择性质。指问题的整体最优解可以通过一系列局部最优的选择，即贪心选择来得到。这是贪心法和动态规划法的主要区别。

8.5.2　典型例题分析

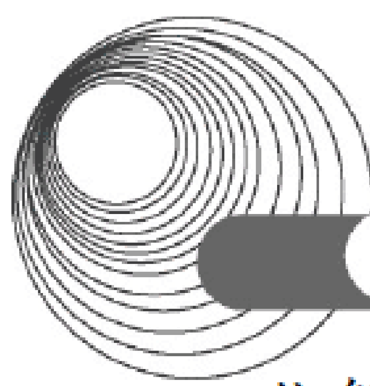
例 1　考虑下述背包问题的实例。有 5 件物品，背包容量为 100，每件物品的价值和重量如下表所示，并已经按照物品的单位重量价值从大到小排好序。根据物品单位重量价值大优先的策略装入背包中，则采用了 (60) 设计策略。考虑 0/1 背包问题(每件物品或者全部装入背包或者不装入背包)和部分背包问题(物品可以部分装入背包)，求解该实例得到的最大价值分别为 (61) 。(2013 年上半年试题 60、61)

物品编号	价值	重量
1	50	5
2	200	25
3	180	30
4	225	45
5	200	50

(60) A.分治　B. 贪心　C. 动态规划　D. 回溯
(61) A.605 和 630　B. 605 和 605　C. 430 和 630　D. 630 和 430

解析：本题考查贪心算法和背包问题的知识点。

贪心算法(又称贪婪算法)是指，在对问题求解时，总是作出在当前看来是最好的选择。也就是说，不从整体最优上加以考虑，他所作出的仅是在某种意义上的局部最优解。贪心算法不是对所有问题都能得到整体最优解，但对范围相当广泛的许多问题他能产生整体最



优解或者是整体最优解的近似解。

采用 0/1 背包考虑该问题时，只能放入 1、2、3 号物品，故总价值为 430，采用部分背包可以将物品拆分，故放入 1、2、3 号物品后还可以将编号 4 的物品部分地装入，使得背包容量尽量满，故总容量为 630。

答案：(60) B　(61) C

例 2　霍夫曼编码将频繁出现的字符采用短编码，出现频率较低的字符采用长编码。具体的操作过程为：①以每个字符的出现频率作为关键字构建最小优先级队列；②取出关键字最小的两个节点生成子树，根节点的关键字为孩子节点关键字之和，并将根节点插入到最小优先级队列中，直至得到一棵最优编码树。霍夫曼编码方案是基于__(64)__策略的，用该方案对包含 a～f 这 6 个字符的文件进行编码，文件包含 100 000 个字符，每个字符的出现频率(用百分比表示)如下表所示，则与固定长度编码相比，该编码方案节省了__(65)__存储空间。(2012 年下半年试题 64、65)

字符	a	b	c	d	e	f
出现频率	18	32	4	8	12	26

(64) A．分治　B．贪心　C．动态规划　D．回溯

(65) A．21%　B．27%　C．18%　D．36%

解析：贪心算法在解决最优化问题上是仅根据当前已有的信息作出选择，即不是从整体最优考虑，它所作出的选择只是力求局部最优。本题给出的霍夫曼编码操作过程基于典型的贪心策略。

采用固定长度编码，需要 3 位二进制数字来表示 6 个字符，即 a=000，b=001，c=010，d=011，e=100，f=101。这种方法需要 300 000 位来对整个原文件编码。采用霍夫曼编码，频繁出现的字符采用短编码，出现频率较低的字符采用长编码，这种编码方式需要(32×1+26×3+18×3+12×3+4×4+8×4)×1000=248000 位。因此与固定长度编码相比，该编码方案节省的存储空间为(300000−248000)/300000 = 17.3%。

答案：(64) B　(65) C

例 3　某货车运输公司有一个中央仓库和 n 个运输目的地，每天要从中央仓库将货物运输到所有运输目的地，到达每个运输目的地一次且仅一次，最后回到中央仓库。在两个地点 i 和 j 之间运输货物存在费用 C_{ij}，为求解旅行费用总和最小的运输路径，设计以下算法：首先选择离中央仓库最近的运输目的地 1，然后选择离运输目的地 1 最近的运输目的地 2，……，每次在来访问过的运输目的地中选择离当前运输目的地最近的运输目的地，最后回到中央仓库，则该算法采用了__(63)__算法设计策略，其时间复杂度为__(64)__。(2012 年上半年试题 63、64)

(63) A. 分治　B. 动态规划　C. 贪心　D. 回溯

(64) A. $\Theta(n^2)$　B. $\Theta(n)$　C. $\Theta(n\lg n)$　D. $\Theta(1)$

解析：贪心算法不考虑整体情况，仅以当前情况为基础作出最优选择。很明显，题目中用到的是贪心算法。分治算法是将规模为 n 的问题分解为 k 个子问题，这些子问题相互独立，且与原问题相同，然后将子问题的解合并得到原问题的解。动态规划算法与分治算法类似，但分解后的子问题往往不是独立的。回溯法要在包含问题的所有解的解空间中，

按照深度优先的策略，从根节点出发搜索解空间。

在选择路径时，首先选择离中央仓库最近的运输目的地 1，需要将所有 n 个目的地到中央仓库的距离进行比较，选择最近的作为目的地 1，相当于从 n 个数中选择一个最小数，此时比较了 $n-1$ 次；然后选择离目的地 1 最近的目的地 2，此时需要将其余 $n-1$ 个目的地到目的地 1 的距离进行比较，相当于从 $n-1$ 个数中选择一个最小数，此时比较了 $n-2$ 次，以此类推，共需比较 $n-1+n-2+n-3+\dots+2+1=(n-1)(n-2)/2=(n^2-3n+2)/2$，算法的时间复杂度为 $\Theta(n^2)$。

答案：(63) C　　(64) A

例 4　迪杰斯特拉(Dijkstra)算法用于求解图上的单源点最短路径。该算法按路径长度递增次序产生最短路径，从本质上说，该算法是一种基于 (62) 策略的算法。(2011 年下半年试题 62)

(62) A. 分治　　B. 动态规划　　C. 贪心　　D. 回溯

解析：Dijkstra 用来解决从顶点 v_0 出发到其余顶点的最短路径，该算法按照最短路径长度递增的顺序产生所有最短路径：对于图 $G=(V, E)$，将图中的顶点分成 S 和 T 两组，S 为已求出的最短路径的终点集合(开始为 $\{v_0\}$)，T 为尚未求出最短路径的终点集合(开始为 $V-\{v_0\}$ 的全部节点)。算法将按最短路径长度的递增顺序逐个将 T 的顶点加入到 S 中，直到所有顶点都被加入到顶点集 S 为止。本质上说，该算法是一种基于贪心策略的算法。贪心算法根据当前已有的信息作出选择，所作出的选择是局部上的最优。

答案：C

例 5　(65) 不能保证求得 0-1 背包问题的最优解。(2010 年下半年试题 65)

(65) A. 分支-限界法　　B. 贪心算法
　　C. 回溯法　　D. 动态规划策略

解析：贪心法在解决问题的策略上仅根据当前已有的信息作出选择，而且一旦作出了选择，不管将来有什么结果，这个选择都不会改变。也就是说，贪心法并不是从整体最优考虑，它所作出的选择只是在某种意义上的局部最优。这种局部最优选择并不能保证总能获得全局最优解，但通常能得到较好的近似最优解。

答案：B

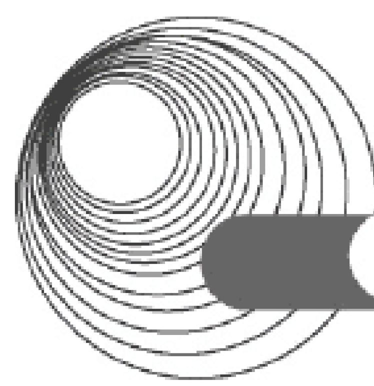
8.5.3　同步练习

1. 迪杰斯特拉(Dijkstra)算法按照路径长度递增的方式求解单源点最短路径问题，该算法运用了______算法策略。

A. 贪心　　B. 分而治之　　C. 动态规划　　D. 试探＋回溯

2. 设商店有 10、5、2 和 1 元的零币，每种零币的数量充足。售货员给顾客找零钱时，零币的数量越少越好。例如，给顾客找零 29 元：先选择两张 10 元币，然后选择一张 5 元币，再选择两张 2 元币。以上的找零钱方法采用了______策略。

A. 分治　　B. 贪心　　C. 动态规划　　D. 回溯



8.5.4 同步练习参考答案

1. A　　　　　2. B

8.6 回　溯　法

8.6.1 考点辅导

回溯法也称为试探法，该方法首先暂时放弃关于问题规模大小的限制，并将问题的候选解按某种顺序逐一枚举和检验。当发现当前候选解不可能是解时，就选择下一个候选解；倘若当前候选解除了还不满足问题规模要求外，满足所有其他要求时，继续扩大当前候选解的规模，并继续试探。如果当前候选解满足包括问题规模在内的所有要求时，该候选解就是问题的一个解。在回溯法中，放弃当前候选解，寻找下一个候选解的过程称为回溯；扩大当前候选解的规模，以继续试探的过程称为向前试探。

应用回溯法解问题时，首先应明确定义问题的解空间。问题的解空间应至少包含问题的一个(最优)解。

确定了解空间的组织结构后，回溯法从开始节点(根节点)出发，以深度优先的方式搜索整个解空间。这个开始节点就称为一个活节点，同时也称为当前的扩展节点。在当前的扩展节点处，搜索向纵深方向移至一个新节点。这个新节点就成为一个新的活节点，并成为当前扩展节点。如果在当前扩展节点处不能再向纵深方向移动，则当前的扩展节点就成为死节点。换句话说，这个节点不再是一个活节点。此时，应往回移动(回溯)至最近的一个活节点处，并使这个活节点成为当前的扩展节点。回溯法即以这种工作方式递归地在解空间中搜索，直至找到所要求的解或解空间中已无活节点时为止。

8.6.2 典型例题分析

例 1　要在 8×8 的棋盘上摆放 8 个“皇后”，要求“皇后”之间不能发生冲突，即任何两个“皇后”不能在同一行、同一列和相同的对角线上，则一般采用(62)来实现。(2011 年上半年试题 62)

(62) A．分治法　　B．动态规划法　　C．贪心法　　D．回溯法

解析：8 皇后问题等价于要求在一个 8×8 的棋盘上放置 8 个皇后，使得任意两个皇后不能放在同一行或同一列或同一斜线上。求解过程从空棋盘开始，设在第 1 行至第 m 行都已经正确放置了 m 个皇后的基础上，再在第 m+1 行上找合适的位置放置第 m+1 个皇后，直至第 8 行也找到合适的位置放置第 8 个皇后。在任一行上都有 8 种选择，开始时，位置在第 1 列，以后改变时，顺序选择第 2 列、第 3 列、……、第 8 列。当第 8 列也不是一个合适的位置时，就要回溯，去改变前一行的位置。

分治法将复杂的大问题分解成规模小的问题以各个击破。归并排序等算法用到的是分

治法实现。动态规划法与分治法类似，其基本思想也是将待求解问题分解成若干子问题，先求解子问题，然后从这些子问题的解得到原问题的解，背包问题、LCS 问题等是采用动态规划法实现的。贪心法跟动态规划法一样，也是用来解决最优问题的，但贪心法并不从整体最优考虑，它所作出的选择只是某种意义上的局部最优。

答案：D

例 2　以下的算法设计方法中，(64) 以获取问题最优解为目标。(2009 年上半年试题 64)

(64) A．回溯法　　　B．分治法　　　C．动态规划法　　　D．递推法

解析：在此做个各种算法基本思想的总结，了解各种算法的基本思想后再解答这个问题就很简单了。

回溯法也称为试探法，该方法首先暂时放弃关于问题规模大小的限制，并将问题的候选解按某种顺序逐一枚举和检验。当发现当前候选解不可能是解时，就选择下一个候选解；倘若当前候选解除了还不满足问题规模要求外，已满足所有其他要求时，继续扩大当前候选解的规模，并继续试探。如果当前候选解满足包括问题规模在内的所有要求时，该候选解就是问题的一个解。在回溯法中，放弃当前候选解，寻找下一个候选解的过程称为回溯。扩大当前候选解的规模，以继续试探的过程称为向前试探。

分治法的设计思想是，将一个难以直接解决的大问题，分割成一些规模较小的相同问题，以便各个击破、分而治之。

动态规划法：经常会遇到复杂问题不能简单地分解成几个子问题，而会分解出一系列的子问题。简单地采用把大问题分解成子问题，并综合子问题的解导出大问题的解的方法，问题求解耗时会按问题规模成幂级数增加。为了节约重复求相同子问题的时间，引入一个数组，不管它们是否对最终解有用，把所有子问题的解存于该数组中，这就是动态规划法所采用的基本方法。

适用动态规划的问题必须满足下列条件。

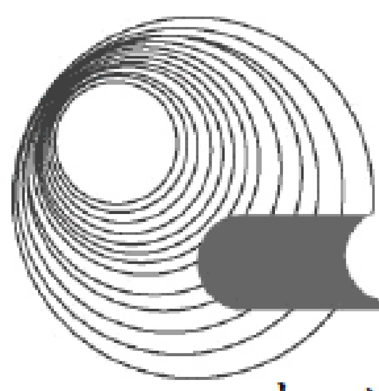
(1)　最优化原理(最优子结构性质)。最优化原理可这样阐述：一个最优化策略具有这样的性质，不论过去状态和决策如何，对前面的决策所形成的状态而言，余下的诸决策必须构成最优策略。简而言之，一个最优化策略的子策略总是最优的。一个问题满足最优化原理又称其具有最优子结构性质。

最优化原理是动态规划的基础，任何问题，如果失去了最优化原理的支持，就不可能用动态规划方法计算。根据最优化原理导出的动态规划基本方程是解决一切动态规划问题的基本方法。

(2)　无后向性。将各阶段按照一定的次序排列好之后，对于某个给定的阶段状态，它以前各阶段的状态无法直接影响它未来的决策，而只能通过当前的这个状态。换句话说，每个状态都是过去历史的一个完整总结。这就是无后向性，又称为无后效性。

(3)　子问题的重叠性。动态规划算法的关键在于解决冗余，这是动态规划算法的根本目的。动态规划实质上是一种以空间换时间的技术，它在实现的过程中，不得不存储产生过程中的各种状态，所以它的空间复杂度要大于其他的算法。选择动态规划算法是因为动态规划算法在空间上可以承受，而搜索算法在时间上却无法承受，所以舍空间而取时间。

所以，能够用动态规划法解决的问题还有一个显著特征：子问题的重叠性。这个性质并不是动态规划法适用的必要条件，但是如果该性质无法满足，动态规划算法同其他算法



相比就不具备优势。

递推法是利用问题本身所具有的一种递推关系求问题解的一种方法。设要求问题规模为 N 的解，当 N=1 时，解或为已知，或能非常方便地得到解。能采用递推法构造算法的问题有重要的递推性质，即当得到问题规模为 i−1 的解后，由问题的递推性质，能从已求得的规模为 1，2，…，i−1 的一系列解，构造出问题规模为 I 的解。这样，程序可从 i=0 或 i=1 出发，重复地，由已知至 i−1 规模的解，通过递推，获得规模为 i 的解，直至得到规模为 N 的解。

迭代法是用于求方程或方程组近似根的一种常用的算法设计方法。

穷举搜索法是对可能是解的众多候选解按某种顺序进行逐一枚举和检验，并从中找出那些符合要求的候选解作为问题的解。

递归是设计和描述算法的一种有力的工具，它在复杂算法的描述中被经常采用。能采用递归描述的算法通常有这样的特征：为求解规模为 N 的问题，设法将它分解成规模较小的问题，然后从这些小问题的解方便地构造出大问题的解，并且这些规模较小的问题也能采用同样的分解和综合方法，分解成规模更小的问题，并从这些更小问题的解构造出规模较大问题的解。特别地，当规模 N=1 时，能直接得解。

贪心法是一种不追求最优解，只希望得到较为满意解的方法。贪心法一般可以快速得到满意的解，因为它省去了为找最优解要穷尽所有可能而必须耗费的大量时间。贪心法常以当前情况为基础作最优选择，而不考虑各种可能的整体情况，所以贪心法不需要回溯。

答案：A

8.6.3 同步练习

在分支-限界算法设计策略中，通常采用_____搜索问题的解空间。

A．深度优先　　B．广度优先　　C．自底向上　　D．拓扑序列

8.6.4 同步练习参考答案

B

8.7 分支限界法

注：此节内容不是考试重点，考生了解即可。

分支限界法类似于回溯法，也是一种在问题的解空间树上搜索问题解的算法。但在一般情况下，分支限界法与回溯法的求解目标不同。回溯法的求解目标是找出解空间树中满足约束条件的所有解，而分支限界法的求解目标则是找出满足约束条件的一个解，或是在满足约束条件的解中找出使某一目标函数值达到极大或极小的解，即在某种意义下的最优解。由于求解目标不同，导致分支限界法与回溯法在解空间树上的搜索方式也不相同。回溯法以深度优先的方式搜索解空间树，而分支限界法则以广度优先或以最小耗费优先的方

式搜索解空间树。分支限界法的搜索策略是，每一个活节点只有一次机会成为扩展节点。活节点一旦成为扩展节点，就一次性产生其所有儿子节点。在这些儿子节点中，那些导致不可行解或非最优解的儿子节点被舍弃，其余儿子节点被加入到活节点表中。此后，从活节点表中取下一节点成为当前扩展节点，并重复上述节点扩展过程。这个过程一直持续到找到所需的解或活节点表为空时为止。

从活节点表中选择下一扩展节点的不同方式导致不同的分支限界法。最常用的有队列式分支限界法和优先队列分支限界法。

8.8　概 率 算 法

注：此节内容不是考试重点，近年来也未曾考过，考生了解即可。

概率算法的一个基本特征是对所求解问题的同一实例用同一概率算法求解两次，可能得到完全不同的效果。这两次求解所需的时间甚至所得到的结果可能会有相当大的差别。

1. 基本特征

一般情况下，概率算法具有以下基本特征。

(1) 概率算法的输入包括两部分：一部分是原问题的输入；另一部分是一个供算法进行随机选择的随机数序列。

(2) 概率算法在运行过程中，包括一处或多处随机选择，根据随机值来决定算法的运行。

(3) 概率算法的结果不能保证一定是正确的，但能限制其出错概率。

(4) 概率算法在不同的运行过程中，对于相同的输入实例可以有不同的结果，因此，对于相同的输入实例，概率算法的执行时间可能不同。

2. 几种类型

一般情况下，可将概率算法大致分为数值概率算法、蒙特卡罗算法、拉斯维加斯算法和舍伍德算法 4 类。

(1) 数值概率算法常用于数值问题的求解。这类算法得到的往往是近似解，且近似解的精度随计算时间的增加不断提高。在多数情况下，要计算出问题的精确解是不可能的或没有必要的，因此数值概率算法可得到相当满意的解。

(2) 蒙特卡罗算法用于求问题的精确解。用蒙特卡罗算法求得问题的一个解，但这个解未必是正确的。求得正确解的概率依赖于算法所用的时间。算法所用的时间越多，得到正确解的概率就越高。蒙特卡罗算法的缺点也在于此。一般情况下，无法有效地判定所得到的解一定正确。

(3) 拉斯维加斯算法不会得到不正确的解。一旦用拉斯维加斯算法找到一个解，这个解一定是正确解。拉斯维加斯算法找到正确解的概率随着它所用的计算时间的增加而提高。对于所求解问题的任一实例，用同一拉斯维加斯算法反复对该实例求解足够多次，可使求解失效的概率任意小。

(4) 舍伍德算法总能求得问题的一个解，且所求得的解总是正确的。当一个确定性算

法在最坏情况下的计算复杂度与其在平均情况下的计算复杂度有较大差别时，可在这个确定性算法中引入随机性将它改造成一个舍伍德算法，消除或减少问题的好坏实例间的这种差别。舍伍德算法的精髓不是避免算法的最坏情况行为，而是设法消除这种最坏情形行为与特定实例之间的关联性。

8.9 近似算法

注：本节不是考试重点，考生了解即可。

近似算法是解决难解问题的一种有效策略，其基本思想是放弃求最优解，而用近似最优解代替最优解，以换取算法设计上的简化和时间复杂度的降低。近似算法是这样一个过程：虽然它可能找不到一个最优解，但它总会为待求解的问题提供一个解。为了具有实用性，近似算法必须能够给出算法所产生的解与最优解之间的差别或者比例的一个界限，它保证任意一个实例的近似最优解与最优解之间相差的程度。显然，这个差别越小，近似算法越具有实用性。

衡量近似算法性能最重要的标准有以下两个。

(1) 算法的时间复杂度。近似算法的时间复杂度必须是多项式阶的，这是近似算法的基本目标。

(2) 解的近似程度。近似最优解的近似程度也是设计近似算法的重要目标。近似程度与近似算法本身、问题规模乃至不同的输入实例有关。

8.10 数据挖掘算法

1. 数据挖掘概述

在当今的大数据时代，数据挖掘、机器学习和人工智能这些名词在我们的生活、工作和学习中已经是耳熟能详的词汇。我们需要各种技术来分析爆炸式增长的各类数据，以发现隐含在这些数据中的有价值的信息和知识。作为一门交叉学科，数据挖掘利用机器学习方法对多种数据，包括数据库数据、数据仓库数据、Web 数据等进行分析和挖掘。数据挖掘的核心是算法，其主要功能包括分类、回归、关联规则和聚类等。

2. 分类

分类是一种有监督的学习过程，根据历史数据预测未来数据的模型。分类的数据对象属性分为两类，一般属性和分类属性或者目标属性。对数据分类有两个步骤：学习模型和应用模型，在分类过程中，涉及到的数据包括训练数据集、测试数据集和未知数据。学习模型是指基于训练数据集采用分类算法建立学习模型。而应用模型是指应用测试数据集的数据到学习模型中，根据输出来评估模型的好坏以及将未知数据输入到学习模型中，预测数据的类型。

存在多种分类算法。决策树归纳是一种自顶向下的递归树算法，使用一种属性选择度

量为树的每个非叶子节点选择待分裂的属性。ID3、C4.5 和 CART 是典型的决策树算法，它们使用不同的属性选择度量。朴素贝叶斯算法和贝叶斯信念网络基于后验概率的贝叶斯公式进行分类，前者假设类条件独立，即数据对象的各个属性之间互相独立，后者考虑属性之间的关系。后向传播(BP)算法是使用梯度下降法的神经网络方法。它搜索一组权重，对数据建模，使得数据对象的预测类型和实际类型之间的平均平方距离最小。支持向量机(SVIVI)是一种用于线性和非线性数据的分类算法。它把输入数据变换到较高维空间，使用称作支持向量的基本元组，从中发现分离数据的超平面。

可以用混淆矩阵来评估分类模型的质量。如对于两类问题，混淆矩阵给出真正例(True Positive)、真负例(True Negative)、假正例(False Positive)、假负例(False Negative)。基于这些量可以计算分类模型的准确率、灵敏度(召回率)、特效性、精度、F 度量等。可以用显著性检验和 ROC 曲线来评价不同分类模型的好坏。

把已知类别的数据集分为训练集和测试集时，可以采用保持、随机抽样、交叉检验和自助法。而可以将多个分类模型组合起来以提高分类的质量，其中袋装、提升和随机森林是典型的组合分类方法。

3. 频繁模式和关联规则挖掘

挖掘海量数据中的频繁模式和关联规则可以有效地指导企业发现交叉销售机会、进行决策分析和商务管理等。一个典型的应用是购物篮分析，即顾客经常购买的商品集合，从而分析顾客的购买习惯。而其中，沃尔玛公司对其顾客购买数据进行分析时，发现购买尿布的客户通常也会购买啤酒，根据这一规律，他们在摆放货架时，有意地把啤酒和尿布放到一起，以便顾客购买。这就是有名的啤酒尿布故事。频繁模式时频繁出现在数据集中的模式，要求满足最小支持度阈值，如啤酒和尿布频繁地出现在同一购物篮中。关联规则是形如 A→B 的规则，其中 A 和 B 表示数据集中的子集，要求 AB 既要满足最小支持度阈值，还要满足最小置信度阈值。如同时购买啤酒和尿布的购物篮满足最小支持度阈值，同时 A→B 满足最小置信度阈值。支持度和置信度的定义如下：

Support(A→B) = P(A U B)

Confidence (A→B) = P(BIA)

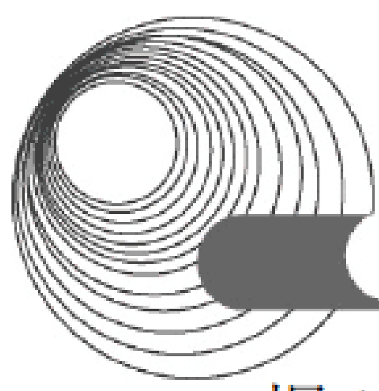
求解关联规则首先要求出数据集中的频繁模式，然后由频繁模式产生关联规则。

人们提出了多种关联规则挖掘算法：类 Apriori 算法；基于频繁模式增长的方法，如 FP-growth；使用垂直数据格式的算法，如 ECLAT。

4. 聚类

聚类是一种无监督学习过程。根据数据的特征，将相似的数据对象归为一类，不相似的数对象归到不同的类中，这就是聚类，每个聚类也称为簇。“物以类聚，人以群分”就是聚类的典型描述。

聚类的典型算法有：基于划分的方法、基于层次的方法、基于密度的方法、基于网格的方法和基于统计模型的方法。基于划分的方法将单个数据对象划分为 k 个不相交的集合，每个集合称为一个簇。典型的算法有 k-均值、k-中心点算法等。基于层次的方法将数据对象集进行层次的分解。根据其是自底向上还是自顶向下分解，可以分为凝聚的方法和分裂的方法，而前者的典型算法是 AGNES，后者的典型算法是 DIANA。基于密度的方法基于数



据对象的邻域来进行聚类分析，因此可以识别各种形状的簇，以及一个数据对象可以属于多个不同的簇，DBSCAN、OPTICS 和 DENCLUE 是其中的典型算法。基于网格的方法把对象空间量化为有限个单元，形成一个网格结构。所有的聚类操作在该网格上进行，STING 和 CLIQUE 是其中的两个算法。基于统计模型的算法将数据对象集看作多个服从不同分布的数据集构成，聚类的目的是识别出这些不同的分布的数据对象，EM 算法是其中的一个典型算法。

5. 数据挖掘的应用

数据挖掘在多个领域已有成功的应用。在银行和金融领域，可以进行贷款偿还预测和顾客信用政策分析、针对定向促销的顾客分类与聚类、洗黑钱和其他金融犯罪侦破等；在零售和电信业，可以进行促销活动的效果分析、顾客忠诚度分析、交叉销售分析、商品推荐、欺骗分析等。

8.11 智能优化算法

优化技术是一种以数学为基础，用于求解各种工程问题优化解的应用技术。作为一个重要的科学分支，它一直受到人们的广泛重视，并在诸多工程领域得到迅速推广和应用，如系统控制、人工智能、模式识别、生产调度、VLSI 技术和计算机工程等。鉴于实际工程问题的复杂性、约束性、非线性、多极性、建模困难等特点，寻求一种适合于大规模并行且具有智能特征的算法已成为有关学科的一个主要研究目标和引人注目的研究方向。20 世纪 80 年代以来，一些新颖的优化算法，如人工神经网络、混沌、遗传算法、进化规划、模拟退火、禁忌搜索及其混合优化策略等，通过模拟或揭示某些自然现象或过程而得到发展，其思想和内容涉及数学、物理学、生物进化、人工智能、神经科学和统计力学等方面，为解决复杂问题提供了新的思路和手段。这些算法独特的优点和机制，引起了国内外学者的广泛重视并掀起了该领域的研究热潮，且在诸多领域得到了成功应用。在优化领域，由于这些算法构造的直观性与自然机理，因而通常被称作智能优化算法，或称现代启发式算法。

1. 人工神经网络

人工神经网络(ANN)是一个以有向图为拓扑结构的动态系统，它通过对连续或断续的输入作状态响应而进行信息处理。人工神经网络技术与计算机技术的结合，为人类进一步研究模拟人类智能及了解人脑思维的奥秘开辟了一条新途径。

2. 遗传算法

遗传算法是源于模拟达尔文的“优胜劣汰、适者生存”的进化论和孟德尔・摩根的遗传变异理论，在迭代过程中保持已有的结构，同时寻找更好的结构。其本意是在人工适应系统中设计一种基于自然的演化机制。

3. 模拟退火算法

模拟退火算法(SA)是一种求解全局优化算法。模拟退火算法的思想是：先将固体加热

至熔化，再让其徐徐冷却，凝固成规整晶体。在加热固体时，固体内部的粒子随着温度的升高，粒子排列从较有序的结晶状态转变为无序的液态，这个过程称为熔解，此时内能增大；冷却时，液体粒子随着温度的徐徐降低，粒子渐趋有序，液体凝固成固体的晶态，这个过程称为退火。最后在常温时达到基态，内能减为最小。

4. 禁忌搜索算法

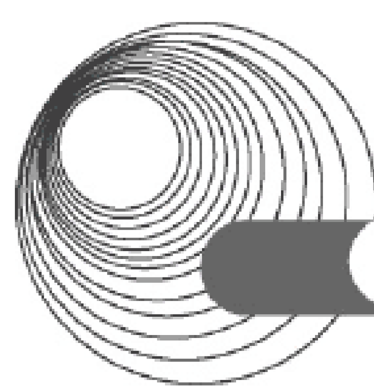
禁忌搜索算法(TS)是模拟人类智力过程的一种全局搜索算法，是对局部邻域搜索的一种扩展。禁忌包含两个方面的意思，一方面，当沿着产生相反结果的道路走下去时，也不会陷入一个圈套而导致无处可逃，另一方面，在必要情况下，保护措施允许被淘汰，也就是说，某种措施被强制运用时，禁忌条件就宣布无效。

禁忌搜索算法是从一个初始可行解出发，选择一系列使目标函数值减少最多(假设求极小值问题)的特定搜索方向(即移动)作为试探，同时为了避免陷入局部最优解，禁忌搜索采用了一种灵活的“记忆”技术，即禁忌表来对已经进行的优化过程进行记录和选择，指导下一步的搜索方向。禁忌表中保存了最近若干次迭代过程中所实现的移动，凡是处于禁忌表中的移动，在当前迭代过程中是不允许实现的，这样可以避免算法重新访问在最近若干次迭代过程中已经访问过的解，从而防止了循环，帮助算法摆脱局部最优解。另外，为了尽可能不错过产生最优解的“移动”，禁忌搜索算法采用藐视准则来赦免一些被禁忌的优良状态，进而保证多样化的有效探索以最终实现全局优化。

5. 蚁群算法

蚁群算法的原理：蚂蚁在寻找食物或者寻找回巢的路径中，会在它们经过的地方留下一些信息素，而信息素能被同一蚁群中后来的蚂蚁感受到，并作为一种信号影响后到者的行动(具体表现在后到的蚂蚁选择有信息素的路径的可能性，比选择没有信息素的路径的可能性大得多)，而后到者留下的信息素会对原有的信息素进行加强，并如此循环下去。这样，经过蚂蚁越多的路径，在后到蚂蚁的选择中被选中的可能性就越大(因为残留的信息素浓度较大)。由于在一定的时间内，越短的路径会被越多的蚂蚁访问，因而积累的信息素也就越多，在下一个时间内被其他的蚂蚁选中的可能性也就越大。这个过程会一直持续到所有的蚂蚁都走最短的那一条路径为止。这种行为表现出一种信息正反馈现象：某一路径上走过的蚂蚁越多，则后到者选择该路径的概率就越大，因此距离近的食物源会吸引越来越多的蚂蚁，信息素浓度的增长速度就会越快，同时通过这种信息的交流，蚂蚁也就寻找到食物与蚁穴之间的最短路径了。

蚁群算法的模型：蚁群算法的主要根据是信息正反馈原理和某种启发式算法的有机结合，其优化过程主要包括选择、更新以及协调 3 个过程。在选择过程中，信息素浓度越高的路径被选择的概率越大；在更新过程中，路径上的信息素随蚂蚁的经过而增长，同时也随时间的推移而挥发；在协调过程中，蚂蚁之间通过信息素进行信息交流相互协作。在选择和更新过程中，较好的解(较短的路径)通过路径上的信息素得到加强，从而引导下一代蚂蚁向较优解邻域搜索使算法收敛，同时更新过程的信息素挥发又使得算法具有探索能力增加解的多样性，使得算法不易陷入局部最优。



6. *粒子群优化算法*

粒子群算法的基本思想：鸟群觅食飞行时，在飞行过程中经常会突然改变方向、散开、聚集，其行为不可预测，但其整体总保持一致性，个体与个体间也保持着最适宜的距离。通过对类似生物群体行为的研究，发现生物群体中存在着一种信息共享机制，为群体的进化提供了一种优势，这就是基本粒子群算法形成的基础。

后来，有学者提出了粒子群优化算法(PSO)。PSO 算法将鸟群运动模型中的栖息地类比为所求问题的解空间中可能解的位置，通过个体间的信息传递，引导整个群体向可能解的方向移动，增加发现较好解的可能性。群体中的鸟被抽象为一个个没有质量、没有形状的“粒子”，通过这些“粒子”的相互协作和信息共享，在解空间中寻找最优解。

8.12 本章小结

本章知识点在 2013 年的新大纲中改动不大，只是有一些描述方面的调整。

本章主要要求考生掌握算法的基础知识、算法与数据结构的关系、算法效率、算法设计，具体的算法包括数值计算方法、递归算法、贪心算法、分治法、动态规划算法等。本章的知识点在上午试题中占的比例比较少，大概 2～3 道小题，每次考到的算法也不一样，所以，大纲要求的算法考生要一一掌握。

本章重点在于对各种算法的掌握，近年考查内容增多，通常与数据结构结合出综合题，详细知识点分析与例题分析将在软件设计师考试同步辅导(下午科目)中详细介绍。

8.13 达标训练题及参考答案

8.13.1 达标训练题

1．用迭代法求解方程 $x^5-x-1=0$，下列迭代公式不可能正确的是______。

A．$x_{k+1}=x_k^5-1$　　B．$x_{k+1}=x_k^{-4}+x_k^{-3}$

C．$x_{k+1}=\sqrt[5]{1+x_k}$　　D．$x_{k+1}=x_k^2-1$

2．以下不属于算法的基本特征的是(1)。穷举法的适用范围是(2)。

(1) A．有确切定义的　　B．可行的

C．可描述的　　D．不能有二义性

(2) A．一切问题　　B．解的个数极多的问题

C．解的个数不太多的问题　　D．不适合设计算法

3．数字三角形问题：

7
3 8
8 1 0
2 7 7 4
5 5 2 6 5

给出了一个数字三角形宝塔。数字三角形中的数字为不超过 100 的正整数。现规定从最顶层走到最底层，每一步可沿左斜线向下或右斜线向下走。假设三角形行数≤100，求解从最顶层走到最底层的一条路径，使得沿着该路径所经过的数字的总和最大，输出最大值。

对于这一问题，很容易想到用(1)的方法去解决。但是当行数很大时，当三角形的行数等于 100 时，用该算法肯定超时，甚至根本不能得到计算结果，所以必须用(2)来解。

(1) A．递归法　　B．穷举法　　C．分治法　　D．动态规划法
(2) A．递推法　　B．回溯法　　C．贪心法　　D．动态规划法

8.13.2　参考答案

1．D
2．(1) C　　(2) C
3．(1) B　　(2) D

第 9 章　数据库技术基础

大纲要求：

- 数据库管理系统的功能和特征。
- 数据库模型(概念模式、外模式、内模式)。
- 数据模型，E-R 图，第一范式、第二范式、第三范式。
- 数据操作(集合运算和关系运算)。
- 数据库语言(SQL)。
- 数据库的控制功能(并发控制、恢复、安全性、完整性等)。
- 数据仓库和分布式数据库基础知识。

9.1　基 本 概 念

9.1.1　考点辅导

9.1.1.1　数据库与数据库管理系统

数据库系统(DataBase System，DBS)从广义上讲是由数据库、硬件、软件和人员组成的，其中管理的对象是数据。数据是经过组织的比特集合，而信息是具有特定释义和意义的数据。

1. 数据库

数据库(DataBase, DB)是指长期存储在计算机内的、有组织的、可共享的数据集合。数据库中的数据按一定的数据模型组织、描述和存储，具有较小的冗余度、较高的数据独立性和易扩展性，并可为各种用户共享。

系统使用的所有数据存储在一个或几个数据库中。

2. 硬件

硬件是指构成计算机系统的各种物理设备，包括存储数据所需的外部设备。

3. 软件

软件包括操作系统、数据库管理系统及应用程序。数据库管理系统(DataBase Management System，DBMS)是数据库系统的核心软件，要在操作系统的支持下工作，解决如何科学地组织和存储数据、如何高效地获取和维护数据库的系统软件问题。其主要功能包括数据定义功能、数据操纵功能、数据库的运行管理和数据库的建立与维护。

4. 人员

与数据库系统有关的人员主要有以下四类。

- 系统分析员和数据库设计人员。
- 应用程序员。
- 最终用户。
- 数据库管理员(DataBase Administrator，DBA)。

9.1.1.2　DBMS 的功能

1. 数据定义

DBMS 提供数据定义语言(Data Description Language，DDL)，用户可以对数据库的结构描述定义，包括外模式、模式和内模式的定义；数据库的完整性定义；安全保密定义。这些定义存储在数据字典中，是 DBMS 运行的基本依据。

2. 数据库操作

DBMS 向用户提供数据操纵语言(Data Manipulation Language，DML)，实现对数据的基本操作，如检索、插入、修改和删除。DML 分为两类，即宿主型和自含型。宿主型是指将 DML 语句嵌入某种主语言中使用；自含型是指可以单独使用 DML 语句，供用户交互使用。

3. 数据库运行管理

数据库在运行期间多用户环境下的并发控制、安全性检查和存取控制、完整性检查和执行、运行日志的组织管理、事务管理和自动恢复等是 DBMS 的重要组成部分。

4. 数据组织、存储和管理

DBMS 分类组织、存储和管理各种数据，包括数据字典、用户数据、存取路径等；要确定以何种文件结构和存取方式在存储级上组织这些数据，以提高存取效率。实现数据间的联系、数据组织和存储的基本目标是提高存储空间的利用率。

5. 数据库的建立和维护

数据库的建立和维护包括数据库的初始建立、数据的转换、数据库的转储和恢复、数据库的重组和重构、性能监测和分析等。

6. 其他功能

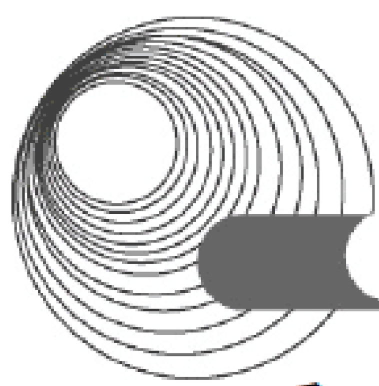
如 DBMS 在网络中与其他软件系统的通信功能，一个 DBMS 与另一个 DBMS 或文件系统的数据转换功能等。

9.1.1.3　DBMS 的特征与分类

1. DBMS 的特征

DBMS 具有以下特征。

(1) 数据结构化且统一管理。数据库中的数据由 DBMS 统一管理。由于数据系统采用复杂的数据模型表示数据结构，数据模型不仅描述数据本身的特点，还描述数据之间的联



系；数据不再面向某个应用，而是面向整个系统模型；数据易维护、易扩展、数据冗余较小，实现了数据共享。

(2) 有较高的数据独立性。数据的独立性是指数据与程序独立，将数据的定义从程序中分离出去，由 DBMS 负责数据的存储，从而简化应用程序，大大减少应用程序编制的工作量。数据的独立性是由 DBMS 的二级映像功能来保证的。数据的独立性包括数据的物理独立性和数据的逻辑独立性。

① 数据的物理独立性是指数据库的内模式发生改变时，数据的逻辑结构不变，从而应用程序不需要修改。

② 数据的逻辑独立性是指用户的应用程序与数据库的逻辑结构是相互独立的。

(3) 数据控制功能。DBMS 提供的数据控制功能包括对数据库中数据的安全性、完整性、并发和恢复的控制。

① 数据的安全性保护。数据的安全性是指保护数据库以防止不合法使用所造成的数据泄露、更改或破坏。

② 数据的完整性。数据的完整性是指数据正确性和相容性，是防止合法用户使用数据库时向数据库加入不符合语义的数据。

③ 并发控制。并发操作带来的问题是数据的不一致性。DBMS 的并发控制子系统负责协调并发事务的执行，保证数据库的完整性不受破坏，避免用户得到不正确的数据。

④ 故障恢复。数据库中的四类故障分别是事务内部故障、系统故障、介质故障及计算机病毒。故障恢复主要是指恢复数据库本身，即在故障引起数据库当前状态不一致后，将数据库恢复到某个正确状态或一致状态。

2. DBMS 的分类

DBMS 通常可分为以下 3 类。

(1) 关系数据库系统(Relation DataBase System，RDBS)是支持关系模型的数据库系统。在关系模型中，实体以及实体间的联系都是用关系来表示。在一个给定的现实世界领域中，所有实体及实体之间的关系的集合所构成的关系数据库也有类型和值之分。关系数据库的型也称为关系数据库模式，是对关系数据库的描述，是关系模式的集合；关系数据库的值也称为关系数据库，是关系的集合。关系数据库模式与关系数据库通常统称为关系数据库。微机方式下的简单的 DBMS 系统包括常见的 FoxBASE、Foxpro、dBASE、Access 等。

(2) 面向对象的数据库系统(Object-Oriented DataBase System，OODBS)是支持以对象形式对数据建模的数据库管理系统。这包括对以下功能的支持：支持对象的类、支持类属性的继承、支持子类。一个面向对象的数据库系统必须符合两个条件：必须是一个 DBMS；必须是面向对象的。

(3) 对象关系数据库系统(Object-Oriented Relation DataBase System，ORDBS)在传统的关系数据模型基础上，提供元组、数组、集合之类更丰富的数据类型以及处理新的数据类型操作的能力，这样形成的数据模型称为对象关系数据模型。基于对象关系数据模型的 DBS 称为对象关系数据库系统。

9.1.1.4 数据库系统的体系结构

站在不同的角度或不同层次上看，数据库系统的体系结构也不同。站在最终用户的角

度看，数据库系统的体系结构分为集中式、分布式、C/S(客户端/服务器)和并行结构。

1. 集中式数据库系统

在集中式数据库系统中，不但数据是集中的，数据的管理也是集中的，数据库系统的所有功能，从形式的用户接口到 DBMS 核心都集中在 DBMS 所在的计算机上。

2. 客户端/服务器体系结构

客户端/服务器结构的数据库系统功能分为前端和后端。前端主要包括图形用户界面、表格生成和报表处理等工具；后端负责存取结构、查询计算和优化、并发控制以及故障恢复等。前端与后端通过 SQL 或应用程序来接口。

数据库服务器一般可分为事务服务器和数据服务器。

3. 并行数据库系统

并行体系结构的数据库系统是多个物理上连在一起的 CPU，而分布式系统是多个地理上分开的 CPU。并行体系结构的数据库类型分为共享内存式多处理器和无共享式并行体系结构。

4. 分布式数据库系统

分布式 DBMS 包括物理上分布、逻辑上集中的分布式数据库结构和物理上分布、逻辑上分布的分布式数据库结构两种。前者的指导思想是把单位的数据模式(称为全局数据模式)按数据来源和用途，合理分布在系统的多个节点上，使大部分数据可以就地或就近存取。数据在物理上分布后，由系统统一管理，使用户不感到数据的分布。后者一般由两部分组成：一是本节点的数据模式；二是本节点共享的其他节点上有关的数据模式。节点间的数据共享由双方协商确定。这种数据库结构有利于数据库的集成、扩展和重新配置。

9.1.1.5　数据库的三级模式结构

1. 模式结构

数据库系统采用三级模式结构，这是数据库管理系统内部的系统结构。

(1) 概念模式(Schema)。也称模式，是数据库中全体数据的逻辑结构和特征的描述，它由若干个概念记录类型组成，只涉及行的描述，不涉及具体的值。概念模式的一个具体值称为模式的一个实例，同一个模式可以有很多实例。

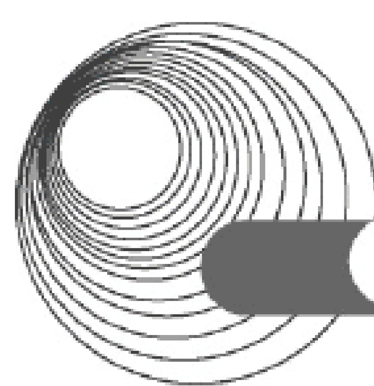
(2) 外模式(External Schema)。也称用户模式或子模式，是用户与数据库系统的接口，是用户用到的那部分数据的描述，由若干个外部记录类型组成。描述外模式的数据定义语言称为外模式 DDL。

(3) 内模式(Internal Schema)。也称存储模式，是数据物理结构和存储方式的描述，是数据在数据库内部的表示方式，定义所有的内部记录类型、索引和文件的组织方式以及数据控制方面的细节。描述内模式的数据定义语言称为内模式 DDL。

2. 两级映像

数据库系统在三级模式之间提供了两级映像，即模式/内模式映像和外模式/模式映像。

(1) 模式/内模式映像。该映像存在于概念级和内部级之间，实现了概念模式到内模式之间的相互转换。



(2) 外模式/模式映像。该映像存在于外部级和概念级之间，实现了外模式到概念模式之间的相互转换。

DBMS 的二级映像功能保证了数据的独立性。

9.1.2 典型例题分析

例 1 在数据库逻辑结构设计阶段，需要 (51) 阶段形成的 (52) 作为设计依据。(2014 年下半年试题 51、52)

(51) A．需求分析　　B．概念结构设计

C．物理结构设计　　D．数据库运行和维护

(52) A．程序文档、数据字典和数据流图。

B．需求说明文档、程序文档和数据流图

C．需求说明文档、数据字典和数据流图

D．需求说明文档、数据字典和程序文档

解析：数据库的设计过程，按照规范的设计方法，一般分为以下 6 个阶段。

(1) 需求分析：分析用户的需求，包括数据、功能和性能需求。

(2) 概念结构设计：主要采用 E-R 模型进行设计，包括画 E-R 图。

(3) 逻辑结构设计：通过将 E-R 图转换成表，实现从 E-R 模型到关系模型的转换。

(4) 数据库物理设计：主要是为所设计的数据库选择合适的存储结构和存取路径。

(5) 数据库的实施：包括编程、测试和试运行。

(6) 数据库运行与维护：系统的运行与数据库的日常维护。

答案：(51) A　(52) C

例 2 在分布式数据库中有分片透明、复制透明、位置透明和逻辑透明等基本概念，其中： (53) 是指局部数据模型透明，即用户或应用程序无需知道局部使用的是哪种数据模型； (54) 是指用户或应用程序不需要知道逻辑上访问的表具体是如何分块存储的。 (2015 年下半年试题 53、54)

(53) A．分片透明　B．复制透明　C．位置透明　D．逻辑透明

(54) A．分片透明　B．复制透明　C．位置透明　D．逻辑透明

解析：分片透明：用户不必关心数据是如何分片的，对数据的操作在全局关系上进行，即关系如何分片对用户是透明的。

复制透明：用户不用关心数据库在网络中各个节点的复制情况，被复制的数据的更新都由系统自动完成。

答案：(53) D　(54) A

例 3 在分布式数据库系统中， (55) 是指用户无需知道数据存放的物理位置。(2013 年下半年试题 55)

(55) A．分片透明　B．复制透明　C．逻辑透明　D．位置透明

解析：分片透明是最高层次的分布透明性，是指用户或应用程序只对全局关系进行操作而不必考虑数据的分片；复制透明性：在分布式系统中，为了提高系统的性能和实用性，有些数据并不只存放在一个场地，很可能同时重复地存放在不同的场地；位置透明性：位

于分片视图与分配视图之间，是指用户或应用程序应当了解分片情况，但不必了解片段的存储场地。所以本题答案选 D。

答案：D

例 4　数据的物理独立性和逻辑独立性分别是通过修改__(51)__来完成的。(2016 年上半年试题 51)

(51) A．外模式与内模式之间的映像、模式与内模式之间的映像

B．外模式与内模式之间的映像、外模式与模式之间的映像

C．外模式与模式之间的映像、模式与内模式之间的映像

D．模式与内模式之间的映像、外模式与模式之间的映像

解析：物理独立性是指内模式发生变化，只需要调整模式与内模式之间的映像，而不用修改应用程序。

逻辑独立性是指模式发生变化，只需要调整外模式与模式之间的映像，而不用修改应用程序。

答案：D

9.2　数 据 模 型

9.2.1　考点辅导

9.2.1.1　数据模型的基本概念

模型就是对现实世界特征的模拟和抽象，数学模型是对现实世界数据特征的抽象。从事物的客观特性到计算机里的具体表示经历了现实世界、信息世界和机器世界 3 个数据领域。

(1)　概念数据模型。也称信息模型，是按用户的观点对数据和信息建模，是现实世界到信息世界的第一层抽象，强调其语义表达功能，易于用户理解，是用户和数据库设计人员交流的语言，主要用于数据库设计。这类模型中最著名的是实体联系模型，简称 E-R 模型。

(2)　基本数据模型。它是按计算机系统的观点对数据建模，是现实世界数据特征的抽象，用于 DBMS 的实现。基本的数据模型有层次模型、网状模型、关系模型和面向对象模型(Object Oriented Model，OOM)。

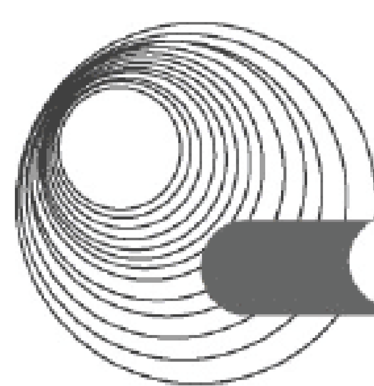
9.2.1.2　数据模型的三要素

数据模型是用来描述数据的一组概念和定义。数据模型的三要素是数据结构、数据操作、数据的约束条件。

(1)　数据结构。它是所研究的对象类型的集合，是对系统静态特性的描述。

(2)　数据操作。它是对数据库中各种对象(型)的实例(值)允许执行的操作集合，包括操作及操作规则。数据操作是对系统动态特性的描述。

(3)　数据的约束条件。这是一组完整性规则的集合。也就是说，对于具体的应用数据必须遵循特定的语义约束条件，以保证数据的正确、有效、相容。



9.2.1.3 E-R 模型

实体–联系模型简称 E-R 模型，所采用的 3 个主要概念是实体、联系和属性。E-R 模型是软件工程设计中的一个重要方法，因为它接近于人的思维方式，容易理解并且与计算机无关，所以用户容易接受。一般遇到实际问题，应先设计一个 E-R 模型，然后再把它转换成计算机能接受的数据模型。

1. 实体

实体是现实世界中可以区别于其他对象的“事件”或“物体”。每个实体由一组特性(属性)来表示，其中的某一部分属性可以唯一表示实体。实体集是具有相同属性的实体集合。

2. 联系

实体集之间的对应关系称为联系。实体的联系分为实体内部的联系和实体与实体之间的联系。实体集内部的联系反映数据在同一记录内部各字段间的联系。而实体集之间的联系类型有一对一联系、一对多联系和多对多联系。

(1) 一对一联系。如果对于实体集 A 中的每一个实体，实体集 B 中至多有一个实体与之联系；反之亦然，则称实体集 A 与实体集 B 具有一对一联系。记为 1∶1。

(2) 一对多联系。如果对于实体集 A 中的每一个实体，实体集 B 中有 n 个实体($n \geqslant 0$)与之联系；反之，对于实体集 B 中的每一个实体，实体集 A 中至多只有一个实体与之联系，则称实体集 A 与实体集 B 有一对多联系。记为 1∶n。

(3) 多对多联系。如果对于实体集 A 中的每一个实体，实体集 B 中有 n 个实体($n \geqslant 0$)与之联系；反之，对于实体集 B 中的每一个实体，实体集 A 中也有 m 个实体($m \geqslant 0$)与之联系，则称实体集 A 与实体集 B 具有多对多联系。记为 m∶n。

多个实体集间的联系类型有多个实体集间的一对一联系、多个实体集间的一对多联系和多个实体集间的多对多联系。

同一个实体集内部的各实体之间也存在 1∶1、1∶n 和 m∶n 的联系。

3. 属性

属性是实体某方面的特性。在同一实体集中，每个实体的属性及其域是相同的，但可能取不同的值。E-R 模型中的属性有以下分类。

(1) 简单属性和复合属性。简单属性是原子的、不可再分的，复合属性可以细分为更小的部分(即划分为别的属性)。

(2) 单值属性和多值属性。若定义的属性对于一个特定的实体只有一个值，这样的属性叫做单值属性；若定义的属性对应一组值，则称为多值属性。

(3) NULL 属性。当实体在某个属性上没有值或属性值未知时，使用 NULL 值，表示无意义或不知道。

(4) 派生属性。可以从其他属性得来。

4. E-R 方法

概念模型中最常用的方法是实体-模型方法，简称 E-R 方法。该方法直接从现实世界中抽象出实体和实体间的联系，然后用非常直观的 E-R 图来表示数据模型。在 E-R 图中有表 9-1 所示的几个主要构件。

表 9-1　E-R 图中的主要构件

构 件 名	构件符号	说　明
矩形		表示实体集
菱形		表示联系集
椭圆		表示属性
线段		将属性与相关的实体集连接，或将实体集与联系集相连
双椭圆		表示多值属性
虚椭圆		表示派生属性
双线		表示一个实体全部参与到联系集中

5. 扩充的 E-R 模型

扩充的 E-R 模型包括弱实体、特殊化、概括、聚集等概念。

9.2.1.4　层次模型

层次模型采用树形结构表示数据与数据间的联系。在层次模型中，每一个节点表示一个记录类型(实体)，记录之间的联系用节点之间的连线表示，并且根节点以外的其他节点有且仅有一个双亲节点。

层次模型不能直接表示多对多的联系。若要表示多对多的联系，可采用以下两种方法。

(1) 冗余节点法。两个实体的多对多的联系转换为两个一对多的联系。该方法的优点是节点清晰，允许节点改变存储位置。缺点是需要额外的存储空间，有潜在的数据不一致性。

(2) 采用虚拟节点分解法，将冗余节点转换为虚拟节点。虚拟节点是一个指引元，指向所代替的节点。该方法的优点是减少对存储空间的浪费，避免数据不一致性。缺点是改变存储位置可能引起虚拟节点中指针的修改。

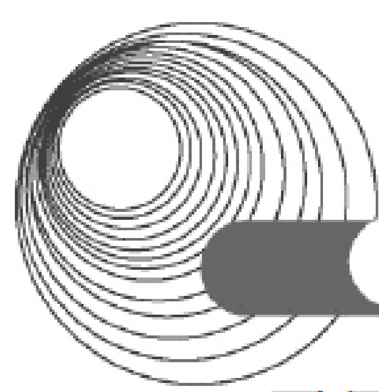
层次模型的优点是：记录之间的联系通过指针实现，比较简单，查询效率高。

层次模型的缺点是：只能表示 $1:n$ 的联系，尽管有许多辅助手段实现 $m:n$ 的联系，但较复杂，不易掌握；由于层次顺序严格和复杂，插入操作的限制比较多，导致应用程序编制比较复杂。

9.2.1.5　网状模型

采用网状结构表示数据与数据间联系的数据模型称为网状模型。在网状模型中，允许一个以上的节点无双亲，一个节点可以有多于一个的双亲。

网状模型是一个比层次模型更具普遍性的数据结构，是层次模型的一个特例。它去掉



了层次模型的两个限制，并允许两个节点之间有多种联系(称之为复合联系)。

网状模型中的每个节点表示一个记录类型(实体)，每个记录类型可以包含若干个字段(实体的属性)，节点间的连线表示记录类型之间一对多的联系。

网状模型在模式 DDL 中提供了定义 DBTG 数据库完整性的若干概念和语句，主要有：支持记录码的概念，保证一个联系中双亲记录和子女记录之间是一对多的联系，支持双亲记录和子女记录之间的某些约束条件。

网状模型的优点是：能够更为直接地描述现实世界，具有良好的性能，存取效率高。

网状模型的缺点是：结构比较复杂。随着应用环境的扩大，数据库的结构变得越来越复杂，不利于最终用户掌握，编制应用程序难度比较大。

9.2.1.6　关系模型

关系数据库系统采用关系模型作为数据的组织方式，在关系模型中用表格结构表达实体集以及实体集之间的联系，其最大特色是描述的一致性。关系模型是由若干个关系模式组成的集合。一个关系模式相当于一个记录型，对应于程序设计语言中类型定义的概念。

关系模型与网状模型、层次模型的最大区别是：用主码而不是用指针导航数据，表格简单、通俗易懂，用户只需要简单地查询语句就可以对数据库进行操作，无须涉及存储结构和访问技术等细节。

关系模型的优点是：概念单一，存储路径对用户是透明的，所以具有更好的数据独立性和安全保密性，简化了程序的开发和数据库的建立工作。

9.2.1.7　面向对象模型

面向对象模型(Object Oriented Model)的核心概念如下。

(1)　对象和对象标识(OID)。对象是现实世界中实体的模型化，与记录、元组的概念相似，但远比它们复杂。每一个对象都有一个唯一的标识，称为对象标识。对象标识不等于关系模式中的记录标识，OID 是独立于值的、全系统唯一的。

(2)　封装(Encapsulate)。每一个对象都是状态和行为的封装。对象的状态是该对象属性的集合，对象的行为是在该对象状态上操作方法(程序代码)的集合。被封装的状态和行为在对象外部是看不见的，只能通过显式定义的消息传递来访问。

(3)　对象的属性(Object Attribute)。对象的属性描述对象的状态、组成和特性，对象的某个属性可以是单值或值的集合。对象的一个属性值本身从该属性的角度看来也是一个对象。

(4)　类和类层次(Class and Class Hierarchy)。

①　类。所有具有相同属性和方法集的对象构成了一个对象类。任何一个对象都是某个对象类的一个实例。对象类中属性的定义域可以是任何类，包括：基本类，如整型、实型和字串等；一般类，包含自身属性和方法类本身。

②　类层次。所有的类组成了一个有根有向无环图，称为类层次(结构)。一个类可以从直接/间接祖先(超类)中继承所有的属性和方法，该类称为子类。

(5)　继承(Inherit)。子类可以从其超类中继承所有属性和方法。类继承可分为单继承(即一个类只能有一个超类)和多重继承(即一个类可以有多个超类)。

9.2.2　典型例题分析

例 1　E-R 模型向关系模型转换时，3 个实体之间多对多的联系 $m:n:p$ 应该转换为一个独立的关系模式，且该关系模式的关键字由__(49)__组成。(2012 年上半年试题 49)

(49) A. 多对多联系的属性　　B. 3 个实体的关键字
　　C. 任意一个实体的关键字　　D. 任意两个实体的关键字

解析：一个 $m:n$ 联系转换为一个关系模式时，与该联系相连的各实体的码以及联系本身的属性均转换为关系的属性。而关系的码为各实体码的组合。3 个或 3 个以上实体间的一个多元联系转换为一个关系模式时，与该多元联系相连的各实体的码以及联系本身的属性均转换为关系的属性，而关系的码为各实体码的组合。

答案：B

例 2　E-R 图转换为关系模型时，对于实体 E_1 与 E_2 间的多对多联系，应该将__(56)__。(2011 年下半年试题 56)

(56) A. E_1 的码加上联系上的属性并入 E_2
　　B. E_1 的码加上联系上的属性独立构成一个关系模式
　　C. E_2 的码加上联系上的属性独立构成一个关系模式
　　D. E_1 与 E_2 的码加上联系上的属性独立构成一个关系模式

解析：E-R 图中有 3 种联系，即一对一、一对多和多对多。在将 E-R 图转换为关系模型时，如果联系是一对一，则可将联系转换成一个独立的关系模式，关系模式的名称取联系的名称，关系模式的属性包括该联系所关联的两个实体的码及联系的属性，关系的码取自任意方实体的码；也可以将联系归并到管理的两个实体的任意方，给待归并的一方实体属性集中增加另一方实体的码和该联系的属性即可，归并后的实体码保持不变。如果联系是一对多，可将联系转换成一个独立的关系模式，关系模式的名称取联系的名称，关系模式的属性包括该联系所关联的两个实体的码及联系的属性，关系的码是多方实体的码；也可将联系归并到关联的两个实体的多方，给待归并的多方实体属性集中增加另一方实体的码和该联系的属性即可，归并后的多方实体码保持不变。如果联系是多对多，则联系只能转换成一个独立的关系模式，关系模式的名称取联系的名称，关系模式的属性取该联系所关联的两个多方的码及联系的属性，关系的码是多方实体的码构成的属性组。

答案：D

例 3　某医院数据库的部分关系模式为：科室(科室号，科室名，负责人，电话)、病患(病历号，姓名，住址，联系电话)和职工(职工号，职工姓名，科室号，住址，联系电话)。假设每个科室有一位负责人和一部电话，每个科室有若干名职工，一名职工只属于一个科室；一个医生可以为多个病患看病；一个病患可以由多个医生多次诊治。科室与职工的所属联系类型是__(51)__，病患与医生的就诊联系类型为__(52)__。对于就诊联系最合理的设计是__(53)__，就诊关系的主键是__(54)__。(2011 年上半年试题 51～54)

(51) A. $1:1$　　B. $1:n$　　C. $n:1$　　D. $n:m$

(52) A. $1:1$　　B. $1:n$　　C. $n:1$　　D. $n:m$

(53) A. 就诊(病历号，职工号，就诊情况)

B．就诊(病历号，职工姓名，就诊情况)

C．就诊(病历号，职工号，就诊时间，就诊情况)

D．就诊(病历号，职工姓名，就诊时间，就诊情况)

(54) A．病历号，职工号　　B．病历号，职工号，就诊时间

C．病历号，职工姓名　　D．病历号，职工姓名，就诊时间

解析：每个科室有若干名职工，一名职工只属于一个科室，所以科室与职工的所属联系类型是一对多，即 1∶*n*。一个医生可以为多个病患看病，一个病患可以由多个医生多次诊治，病患与医生的就诊联系类型为多对多，即 *n*∶*m*。对于不同的职工，职工号是唯一的，不同职工的姓名有可能是相同的，也就是说职工姓名不是唯一的；而不同就诊时间的就诊情况是不同的，因此就诊联系的最合理的设计是就诊(病历号，职工号，就诊时间，就诊情况)，就诊关系的主键是“病历号，职工号，就诊时间”。

答案：(51) B　(52) D　(53) C　(54) B

例 4　给定关系模式 R<U, F>，U={A, B, C}，F={AB→C, C→B}。关系 R__(55)__，且分别有__(56)__。(2011 年上半年试题 55、56)

(55) A．只有 1 个候选关键字 AC　　B．只有 1 个候选关键字 AB

C．有 2 个候选关键字 AC 和 BC　　D．有 2 个候选关键字 AC 和 AB

(56) A．1 个非主属性和 2 个主属性

B．2 个非主属性和 1 个主属性

C．0 个非主属性和 3 个主属性

D．3 个非主属性和 0 个主属性

解析：C→B，AC→AB，而 AB→C，所以 AC→C，关系 R 有两个候选关键字，即 AC 和 AB。包含在任何候选码中的诸属性都是主属性，A、B、C 都包含在候选码中，因此主属性有 3 个，非主属性有 0 个。

答案：(55) D　(56) C

例 5　在某企业的营销管理系统设计阶段，属性“员工”在考勤管理子系统中被称为“员工”，而在档案管理子系统中被称为“职工”，这类冲突称为__(51)__冲突。(2010 年下半年试题 51)

(51) A．语义　B．结构　C．属性　D．命名

解析：题目中，“员工”和“职工”有着相同的意义，但在不同的子系统中有着不同的命名，这就为命名冲突。

如果就同一实体在不同的子系统中有不同的属性，这就为结构冲突。

如果同一属性“员工”在不同的子系统中，属性的类型、取值范围或者数据单位等不一致，这就为属性冲突。

答案：D

例 6　设有学生实体 Students(学号，姓名，性别，年龄，家庭住址，家庭成员，关系，联系电话)，其中“家庭住址”记录了邮编、省、市、街道信息；“家庭成员，关系，联系电话”分别记录了学生亲属的姓名、与学生的关系以及联系电话。

学生实体 Students 中的“家庭住址”是一个__(52)__属性；为使数据库模式设计更合理，对于关系模式 Students__(53)__。(2010 年下半年试题 52、53)

(52) A．简单　　　　B．多值　　　　C．复合　　　　D．派生

(53) A．可以不作任何处理，因为该关系模式达到了 3NF

B．只允许记录一个亲属的姓名、与学生的关系以及联系电话的信息

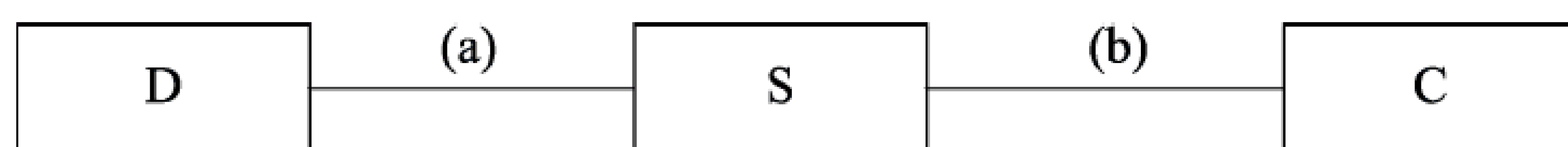
C．需要对关系模式 Students 增加若干组家庭成员、关系及联系电话字段

D．应该将家庭成员、关系及联系电话加上学生号，设计成为一个独立的实体

解析：简单属性是原子的、不可再分的属性，如学号、姓名、性别、年龄等。复合属性则可细分为更小的部分，本题中的家庭地址就是复合属性。属性也有单值属性和多值属性之分，单值属性对于一个特定的实体只有单独的一个值，如学生的学号、姓名、性别、年龄、家庭地址等；而多值属性，一个属性可能对应一组值，如家庭成员。而派生属性，则是从其他属性得来的。

答案：(52) C　　(53) D

例 7　假设有学生 S(学号，姓名，性别，入学时间，联系方式)、院系 D(院系号，院系名称，电话号码，负责人)和课程 C(课程号，课程名)3 个实体，若一名学生属于一个院系，一个院系有多名学生；一名学生可以选择多门课程，一门课程可被多名学生选择，则图中(a)和(b)分别为(51)联系。假设一对多联系不转换为一个独立的关系模式，那么生成的关系模式(52)。(2009 年下半年试题 51、52)



(51) A．1∶*和 1∶*　B．1∶*和*∶1　C．1∶*和*∶*　D．*∶1 和*∶*

(52) A．S 中应加入关系模式 D 的主键

B．S 中应加入关系模式 C 的主键

C．D 中应加入关系模式 S 的主键

D．C 中应加入关系模式 S 的主键

解析：因为一名学生属于一个院系，一个院系有多名学生，因此可知学生 S 和院系 D 是多对一的关系；一名学生可以选择多门课程，一门课程可被多名学生选择，因此学生 S 和课程 C 是多对多的关系；因此第 51 题答案为 C。

一个 1∶*n* 的联系(一对多联系)可转换为一个关系模式，或与 *n* 段的关系模式合并。若独立转换为一个关系模式，那么两端关系的码及其联系的属性为该关系的属性，而 *n* 端的码为关系的码。因此，S 中应加入关系模式 D 的主键。第 52 题答案为 A。

答案：(51) C　　(52) A

例 8　数据库系统通常采用三级模式结构，即外模式、模式和内模式。这三级模式分别对应数据库的 (51) 。(2015 年下半年试题 51)

(51) A．基本表、存储文件和视图　　B．视图、基本表和存储文件

C．基本表、视图和存储文件　　D．视图、存储文件和基本表

解析：外模式又称子模式或用户模式，对应于用户级。它是某个或某几个用户所看到的数据库的数据视图，是与某一应用有关的数据的逻辑表示。

模式又称概念模式或逻辑模式，对应于概念级。它是由数据库设计者综合所有用户的

数据，按照统一的观点构造的全局逻辑结构，是对数据库中全部数据的逻辑结构和特征的总体描述，是所有用户的公共数据视图(全局视图)。它是由数据库管理系统提供的数据模式描述语言(Data Description Language，DDL)来描述、定义的，体现、反映了数据库系统的整体观。

内模式又称存储模式，对应于物理级，它是数据库中全体数据的内部表示或底层描述，是数据库最低一级的逻辑描述，它描述了数据在存储介质上的存储方式和物理结构，对应着实际存储在外存储介质上的数据库。内模式由内模式描述语言来描述、定义，它是数据库的存储观。

答案：B

9.2.3 同步练习

某学校学生、教师和课程实体对应的关系模式如下。

学生(学号，姓名，性别，年龄，家庭住址，电话)

课程(课程号，课程名)

教师(职工号，姓名，年龄，家庭住址，电话)

一个学生可以选修多门课程，一门课程可以有多个学生选修；一个教师只能讲授一门课程，但一门课程可以有多个教师讲授。由于学生和课程之间是一个(1)的联系，所以(2)。又由于教师和课程之间是一个 (3) 的联系，所以 (4) 。

(1) A. 1 对 1　　B. 1 对多　　C. 多对 1　　D. 多对多

(2) A. 不需要增加一个新的关系模式
B. 不需要增加一个新的关系模式，只需要将 1 端的码插入多端
C. 需要增加一个新的选课关系模式，该模式的主键应该为课程号
D. 需要增加一个新的选课关系模式，该模式的主键应该为课程号和学号

(3) A. 1 对 1　　B. 1 对多　　C. 多对 1　　D. 多对多

(4) A. 不需要增加一个新的关系模式，只需要将职工号插入课程关系模式
B. 不需要增加一个新的关系模式，只需要将课程号插入教师关系模式
C. 需要增加一个新的选课关系模式，该模式的主键应该为课程号
D. 需要增加一个新的选课关系模式，该模式的主键应该为课程号和教师号

9.2.4 同步练习参考答案

(1) D　　(2) D　　(3) C　　(4) A

9.3　关 系 代 数

9.3.1　考点辅导

9.3.1.1　关系数据库的基本概念

1. 属性和域

在现实世界中，要描述一个事物，常常取其若干特征来表示。这些特征称为属性。每个属性的取值范围的集合，称为该属性的域。

一般在关系数据库模型中，对域还加了一个限制，所有的域都应是原子数据的集合。关系数据模型的这种限制称为第一范式(1NF)条件。如果关系数据模型突破了 1NF 的限制，则称为非 1NF 的。

2. 笛卡儿积与关系

【定义 9-1】 设 $D_1, D_2, \cdots, D_n$ 为任意集合，定义 $D_1, D_2, \cdots, D_n$ 的笛卡儿积为

$$D_1 \times D_2 \times \cdots \times D_n = \{(d_1, d_2, \cdots, d_n) \mid d_i \in D_i, i = 1, 2, \cdots, n\}$$

其中，每一个元素(d_1，d_2，…，d_n)叫做一个 n 元组，元组的每一个值 d_i 叫做元组的一个分量，若 D_i(i＝1, 2, …, n)为有限集，其基数为 m_i(i＝1, 2, …, n)，则 $D_1 \times D_2 \times \cdots \times D_n$ 的基数 M 为

$$M = \prod_{i=1}^{n} m_i$$

笛卡儿积可以用二维表来表示。

【定义 9-2】 $D_1 \times D_2 \times \cdots \times D_n$ 的子集叫做在域 $D_1, D_2, \cdots, D_n$ 上的关系，记为

$$R(D_1, D_2, \cdots, D_n)$$

称关系 R 为 n 元关系。

由定义 7-2 可以得出，一个关系也可以用二维表来表示。关系中属性的个数称为元数，元组的个数称为基数。

3. 关系的相关名词

下面介绍关系的相关名词。

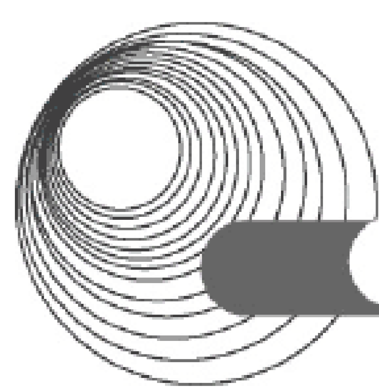
(1) 目或度。常用 R 表示关系的名字，n 表示关系的目或度。

(2) 候选码。若关系中的某一属性或属性组的值能唯一标识一个元组，则称该属性或属性组为候选码。

(3) 主码。若一个关系有多个候选码，则选定其中一个为主码。

(4) 主属性。包含在任何候选码中的诸属性称为主属性。不包含在任何候选码中的属性称为非码属性。

(5) 外码。如果关系模式 R 中的属性或属性组非该关系的码，但它是其他关系的码，那么该属性集对关系模式 R 而言是外码。



(6) 全码。关系模式的所有属性组是这个关系模式的候选码，称为全码。

4. 关系的 3 种类型

下面介绍关系的 3 种类型。

(1) 基本关系(通常又称为基本表、基表)。这是实际存在的表，它是实际存储数据的逻辑表示。

(2) 查询表。查询结果对应的表。

(3) 视图表。这是由基本表或其他视图表导出的表。由于它本身不独立存储在数据库中，数据库中只存放它的定义，所以常称为虚表。

5. 关系数据库模式

在数据库中要区分型和值。关系数据库中的型也称为关系数据库模式，是关系数据库结构的描述，它包括若干域的定义以及在这些域上定义的若干关系模式。关系数据库的值是这些关系模式在某一时刻对应的关系的集合，通常称之为关系数据库。

【定义 9-3】 关系的描述称为关系模式。可以形式化地表示为

$$R(U, D, \mathrm{dom}, F)$$

式中：R 为关系名；U 为组成该关系的属性名集合；D 为属性的域；dom 为属性向域的映像集合；F 为属性间数据的依赖关系集合。

通常将关系模式简记为

$$R(U)\text{或 } R(A_1, A_2, \cdots, A_n)$$

式中：R 为关系名；$A_1, A_2, \cdots, A_n$ 为属性名或域名，属性向域的映像常常直接说明属性的类型、长度。通常在关系模式主属性上加下划线表示该属性为主码属性。

6. 完整性约束

完整性规则提供了一种手段来保证当授权用户对数据库作修改时不会破坏数据的一致性，因此，完整性规则防止的是对数据的意外破坏。关系模型的完整性规则是对关系的某种约束条件。完整性共分为 3 类，即实体完整性、参照完整性(也称引用完整性)和用户自定义完整性。

(1) 实体完整性。规定基本关系 R 的主属性 A 不能取空值。

(2) 参照完整性。现实世界中的实体之间往往存在某种联系，在关系模型中实体与实体之间的联系是用关系来描述的，这样自然就存在着关系与关系间的引用。参照完整性规定，若 F 是基本关系 R 的外码，它与基本关系 S 的主码相对应(基本关系 R 和 S 不一定是不同的关系)，则对于 R 中每个元组在 F 上的值必须为：或者取空值(F 的每个属性值均为空值)，或者等于 S 中某个元组的主码值。

(3) 用户自定义完整性。就是针对某一具体的关系数据库的约束条件，反映某一具体应用所涉及的数据必须满足的语义要求，由应用的环境决定。

7. 关系运算

关系操作的特点是操作对象和操作结果都是集合，而非关系数据模型的数据操作方式则为一次一个记录的方式。关系数据语言分为 3 类，即关系代数语言、关系演算语言和具有关系代数和关系演算双重特点的语言(如 SQL)。关系演算语言包括元组关系演算语言和域

关系演算语言。

关系代数语言、元组关系演算和域关系演算是抽象查询语言，它与具体的 DBMS 中实现的实际语言并不一样，但是可以用它作为评估实际系统中的查询语言能力的标准。

关系运算符有 4 类，即集合运算符、专门的关系运算符、比较算术符和逻辑运算符。关系运算符如表 9-2 所示。

表 9-2　关系运算符

运算符		含义	运算符		含义
集合运算符	$\cup$ $-$ $\cap$ $\times$	并 差 交 笛卡儿积	比较运算符	$>$ $\geqslant$ $<$ $\leqslant$ $=$ $\neq$	大于 大于等于 小于 小于等于 等于 不等于
专门的关系运算符	σ π $\rhd\lhd$ $\div$	选择 投影 连接 除	逻辑运算符	$\neg$ $\wedge$ $\vee$	非 与 或

9.3.1.2　5 种基本的关系代数运算

5 种基本的关系代数运算包括并、差、笛卡儿积、投影、选择，其他运算可以通过基本的关系运算导出。

1. 并

关系 R 与 S 具有相同的关系模式，即 R 与 S 的元数相同(结构相同)。关系 R 和关系 S 的并(Union)由属于 R 或属于 S 的元组构成的集合组成，记作

$$R \cup S = \{t \mid t \in R \vee t \in S\}$$

式中，t 为元组变量。

2. 差

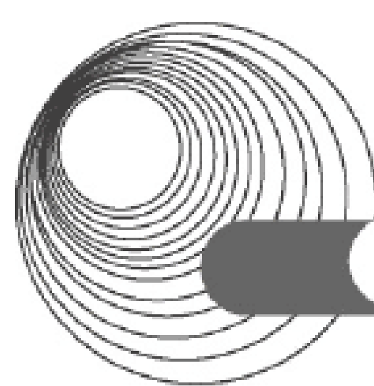
关系 R 与 S 具有相同的关系模式。关系 R 与 S 的差(Difference)由属于 R 但不属于 S 的元组构成的集合组成，记作

$$R - S = \{t \mid t \in R \wedge t \notin S\}$$

3. 广义笛卡儿积

两个元数分别为 n 目和 m 目的关系 R 和 S 的广义笛卡儿积(Extended Cartesian Product)是一个 $n+m$ 列的元组的集合。元组的前 n 列是关系 R 的一个元组，后 m 列是关系 S 的一个元组。若 R 有 k_1 个元组，S 有 k_2 个元组，则关系 R 和 S 的广义笛卡儿积有 $k_1 \times k_2$ 个元组。记作

$$R \times S = \{t \mid t = \langle t_n, t_m \rangle \wedge t_n \in R \wedge t_m \in S\}$$



4. 投影

投影(Projection)运算是从关系的垂直方向进行运算，在关系 R 中选择出若干属性列 A 组成新的关系。记作

$$\pi_A(R) = \{t[A] \mid t \in R\}$$

5. 选择

选择(Selection)运算是从关系的水平方向进行运算，是从关系 R 中选择满足给定条件的诸元素，记作

$$\sigma_F(R) = \{t \mid t \in R \wedge F(t) = \text{true}\}$$

式中，F 中的运算对象是属性名(或列的序号)或常数，运算符是算术比较符和逻辑运算符。

9.3.1.3 扩展的关系代数运算

扩展的关系代数运算可以从基本的关系运算中导出，主要包括以下几种。

1. 交

关系 R 与 S 具有相同的关系模式。关系 R 和 S 的交由属于 R 同时又属于 S 的元组构成的集合组成。关系 R 和 S 的交记作

$$R \cap S = \{t \mid t \in R \wedge t \in S\}$$

显然，$R \cap S = R-(R-S)$或者 $R \cap S = S-(S-R)$。

2. 连接

连接(Join)分为 θ 连接、等值连接和自然连接 3 种。连接运算是从两个关系的笛卡儿积中选取满足条件的元组。

(1) θ 连接。从关系 R 和 S 的笛卡儿积中选取属性间满足一定条件的元组。记作

$$\underset{X\theta Y}{R \bowtie S} = \{t \mid t = \langle t_n, t_m \rangle \wedge t_n \in R \wedge t_m \in S \wedge t_n[X]\theta t_m[Y]\}$$

(2) 等值连接。当 θ 为“=”时称为等值连接。可以表示为

$$\underset{X=Y}{R \bowtie S} = \{t \mid t = \langle t_n, t_m \rangle \wedge t_n \in R \wedge t_m \in S \wedge t_n[X] = t_m[Y]\}$$

(3) 自然连接。是一种比较特殊的等值连接，它要求两个关系中进行比较的分量必须是相同的属性组，并且在结果集中把重复属性列去掉。

3. 除

除(Division)运算是同时从关系的水平方向和垂直方向进行运算。给定关系 $R(X，Y)$和 $S(Y，Z)$，X、Y、Z 为属性组。$R \div S$ 应当满足元组在 X 上的分量值 x 的象集 Y_x 包含关系 S 在属性组 Y 上投影的集合。其形式定义为

$$R \div S = \{t_n[X] \mid t_n \in R \wedge \pi_y(S) \subseteq Y_x\}$$

式中，Y_x 为 x 在 R 中的象集，$x = t_n[X]$，且 $R \div S$ 的结果集的属性组为 X。

4. 广义投影

广义投影(Generalized Projection)运算允许在投影列表中使用算术运算，实现了对投影运

算的扩充。

若有关系 R，条件 $F_1, F_2, \cdots, F_n$ 中的每一个都是涉及 R 中常量和属性的算术表达式，那么广义投影运算的形式定义为

$$\pi_{F_1,F_2,\cdots,F_n}(R)$$

5. 外连接

外连接(Outer Join)运算是连接运算的扩展，可以处理缺失的信息。外连接运算有 3 种，即左外连接、右外连接和全外连接。

(1) 左外连接。取出左侧关系中所有与右侧关系中任一元组都不匹配的元组，用空值 NULL 来填充所有来自右侧关系的属性，构成新的元组，将其加入自然连接的结果中。

(2) 右外连接。取出右侧关系中所有与左侧关系中任一元组都不匹配的元组，用空值 NULL 来填充所有来自左侧关系的属性，构成新的元组，将其加入自然连接的结果中。

(3) 全外连接。完成左外连接和右外连接的操作。即填充左侧关系中所有与右侧关系中任一元组都不匹配的元组，填充右侧关系中所有与左侧关系中任一元组都不匹配的元组，将产生的新元组加入自然连接的结果中。

9.3.2　典型例题分析

例 1　在数据库系统中，视图是一个 (54) 。(2013 年上半年试题 54)

(54) A. 真实存在的表，并保存了待查询的数据

B. 真实存在的表，只有部分数据来源于基本表

C. 虚拟表，查询时只能从一个基本表中导出

D. 虚拟表，查询时可以从一个或者多个基本表或视图中导出

解析：视图是从一个或几个基本表(或视图)导出的虚拟表，其内容由查询定义。同真实的表一样，视图包含一系列带有名称的列和行数据。但是，视图并不在数据库中以存储的数据值集形式存在。行和列数据来自由定义视图的查询所引用的表，并且在引用视图时动态生成。

答案：D

例 2　关系 R1 和 R2 如下图所示。

R1

A	*B*	*C*	*D*
a	d	c	e
c	b	a	e
d	e	c	e
e	f	d	a

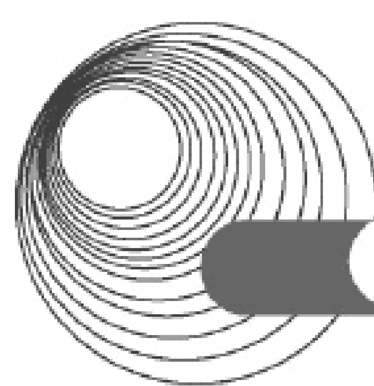
R2

C	*D*	*E*	*F*
a	e	c	a
a	e	a	b
c	e	b	c

若进行 $R1 \bowtie R2$ 运算，则结果集为 (51) 元关系，共有 (52) 个元组。(2012 年下半年试题 51、52)

(51) A. 4　　B. 5　　C. 6　　D. 7

(52) A. 4　　B. 5　　C. 6　　D. 7



解析：$R1\bowtie R2$ 即 $R1$ 与 $R2$ 作自然连接运算，要求两个关系中进行比较的分量必须是相同的属性组，并且在结果集中将重复属性列去掉。进行 $R1\bowtie R2$ 运算后，结果集的属性列应为 *ABCDEF*，即 6 元关系，运算结果如下表所示，共有 4 个元组。

A	*B*	*C*	*D*	*E*	*F*
a	d	c	e	b	c
c	b	a	e	c	a
c	b	a	e	a	B
d	e	c	e	b	C

答案：(51) C　　(52) A

例 3　若对关系 $R(A, B, C, D)$进行$\pi_{1,3}(R)$运算，则该关系运算与 (52) 等价，表示 (53) 。(2012 年上半年试题 52～53)

(52) A. $\pi_{A=1,C=3}(R)$　　B. $\pi_{A=1\wedge C=3}(R)$　　C. $\pi_{A,C}(R)$　　D. $\pi_{A=1\vee C=3}(R)$

(53) A. 属性 *A* 和 *C* 的值分别等于 1 和 3 的元组为结果集

B. 属性 *A* 和 *C* 的值分别等于 1 和 3 的两列为结果集

C. 对 *R* 关系进行 *A*=1、*C*=3 的投影运算

D. 对 *R* 关系进行属性 *A* 和 *C* 的投影运算

解析：$\pi_{1,3}(R)$表示对 *R* 进行投影运算，选择的是 *R* 的第 1 列和第 3 列，也就是 *A* 列和 *C* 列。

答案：(52) C　　(53) D

例 4　若关系 *R*、*S* 如下图所示，则关系代数表达式$\pi 1,3,7(\sigma 3<6(R\times S))$与 (52) 等价。(2010 年上半年试题 52)

A	*B*	*C*	*D*
1	2	4	6
2	3	3	1
3	4	1	3

R

C	*D*	*E*
3	4	2
8	9	3

S

(52) A. $\pi A,C,E(\sigma C<D(R\times S))$　　B. $\pi A,R.C,E(\sigma R.C<S.D(R\times S))$

C. $\pi A,S.C,S.E(\sigma R.C<S.D(R\times S))$　　D. $\pi R.A,R.C,R.E(\sigma R.C<S.D(R\times S))$

解析：本题要求关系代数表达式$\pi_{1,3,7}(\sigma_{3<6}(R\times S))$的结果集，其中，$R\times S$ 的属性列名分别为：*R.A*，*R.B*，*R.C*，*R.D*，*S.C*，*S.D* 和 *S.E*，其结果如下表所示。

R.A	R.B	R.C	R.D	S.C	S.D	S.E
1	2	4	6	3	4	2
1	2	4	6	8	9	3
2	3	3	1	3	4	2
2	3	3	1	8	9	3
3	4	1	3	3	4	2
3	4	1	3	8	9	3
	$R\times S$					

$\sigma_{3<6}(R\times S)$的含义是从 $R\times S$ 结果集中选取第三个分量($R.C$)小于第六个分量($S.D$)的元祖，故$\sigma_{3<6}(R\times S)$与$\sigma_{R.C<S.D}(R\times S)$等价。从上表中可以看出，满足条件的结果如下表所示。

R.A	R.B	R.C	R.D	S.C	S.D	S.E
1	2	4	6	8	9	3
2	3	3	1	3	4	2
2	3	3	1	8	9	3
3	4	1	3	3	4	2
3	4	1	3	8	9	3

$\sigma_{3<6}(R\times S)$

$\pi_{1,3,7}(\sigma_{3<6}(R\times S))$的含义是从$\sigma_{3<6}(R\times S)$结果集中选取第一列 $R.A$(或 A)、第三列 $R.C$ 和第七列 $S.E$(或 E)，故 $\pi_{1,3,7}(\sigma_{3<6}(R\times S))$与 $\pi_{A,R.C,E}(\sigma_{R.C<S.D}(R\times S))$等价。需要说明的是第三列不能简写为 C，因为关系 S 的第一列属性名也为 C，故必须标上关系名加以区别。

答案：B

例 5　设有员工实体 Emp(员工号，姓名，性别，年龄，出生年月，联系方式，部门号)，其中“联系方式”要求记录该员工的手机号码和办公室电话，“部门号”要求参照另一部门实体 Dept 的主码“部门号”。Emp 实体中存在派生属性和多值属性：__(54)__；对属性部门号应该进行__(55)__约束；可以通过命令__(56)__修改表中的数据。(2009 年下半年试题 54～56)

(54) A．年龄和出生年月　　B．年龄和联系方式
　　C．出生年月和联系方式　　D．出生年月和年龄

(55) A．非空主键　　B．主键　　C．外键　　D．候选键

(56) A．INSERT　　B．DELETE　　C．UPDATE　　D．MODIFY

解析：属性可以分为单值属性和多值属性。如果定义的属性对于一个特定的实体都只有单独的一个值，称为单值属性，而在某些特定的情况下，一个属性可能对应一组值。本题中的联系方式包括手机号码和办公室电话两个值，属于多值属性。

而派生属性可以从其他属性得来。本题中的年龄可以由其出生年月得到，因此属于派生属性。

第 55 题中，部门号作为员工实体 Emp 中的外键，因此应该加以外键约束。

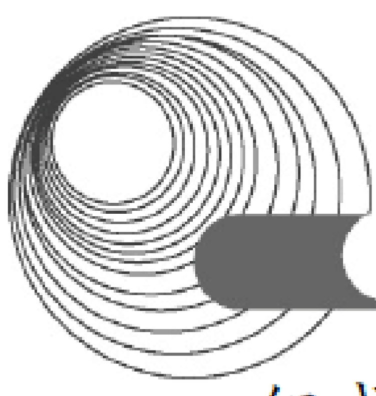
第 56 题，INSERT 是插入命令，DELETE 是删除命令，UPDATE 是更新命令，所以修改表中的数据要通过命令 UPDATE。

答案：(54) B　　(55) C　　(56) C

例 6　若关系 R(H，L，M，P)的主键为全码(All-key)，则关系 R 的主键应__(51)__。(2015 年上半年试题 51)

(51) A．为 HLMP
　　B．在集合{H，L，M，P)中任选一个
　　C．在集合{HL，HM，HP，LM，LP，MP)中任选一个
　　D．在集合{HLM，HLP，HMP，LMP)中任选一个

解析：关系模型的所有属性组成该关系模式的候选码，称为全码。主键应为所有属性



组成的属性组。

答案：A

9.3.3 同步练习

1．关系 R、S 如下所示，关系代数表达式$\pi_{1,5,6}(\sigma_{2=5}(R\times S))$=(1)，该表达式与(2)等价。

关系 R

A	B	C
3	0	3
2	5	6
5	8	9
8	11	12

关系 S

A	B	C
3	10	11
4	11	6
5	10	13
6	11	14

(1)

A.

A	B	C
3	0	3
4	8	9

B.

A	B	C
8	11	6
8	11	14

C.

A	B	C
5	10	11
5	10	13

D.

A	B	C
2	11	6
2	11	14

(2) A. $\pi_{A,B,C}(\sigma_{B=B}(R\times S))$　　B. $\pi_{R.A,R.B,R.C}(\sigma_{R.B=R.B}(R\times S))$

C. $\pi_{R.A,S.B,S.C}(\sigma_{R.B=S.B}(R\times S))a$　　D. $\pi_{R.A,S.B,S.C}(\sigma_{R.B=S.C}(R\times S))$

2. 若关系 R、S 如下图所示，则 R 与 S 自然连接后的属性列数和元组个数分别为<u>(1)</u>；$\pi_{1,4}(\sigma_{3=6}(R\times S))=$<u>(2)</u>。

R

C	D
c	d
g	f

S

(1) A. 4 和 3　　B. 4 和 6　　C. 6 和 3　　D. 6 和 6

(2) A. $\pi_{A,D}(\sigma_{C=D}(R\times S))$　　B. $\pi_{A,R.D}(\sigma_{S.C=R.D}(R\times S))$

C. $\pi_{A,R.D}(\sigma_{R.C=S.D}(R\times S))$　　D. $\pi_{R.A,R.D}(\sigma_{S.C=R.D}(R\times S))$

3. 若某个关系的主码为全码，则该主码应包含______。

A. 单个属性　　B. 两个属性　　C. 多个属性　　D. 全部属性

4. 关系 R、S 如下所示，$R\bowtie S$ 可由<u>(1)</u>基本的关系运算组成，$R\bowtie S$=<u>(2)</u>。

关系 R

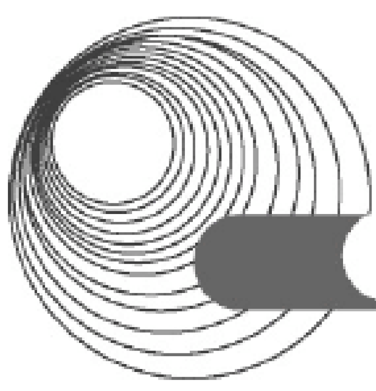
A	B	C
a	b	c
b	a	d
c	d	e
d	f	g

关系 S

A	C	D
a	c	d
d	f	g
b	d	g

(1) A. π、σ 和 $\times$　　B. $-$、σ 和 $\times$

C. $\cap$、σ 和 $\times$　　D. π、σ 和 $\cap$



(2)

A.

A	*B*	*C*
a	b	c
b	a	d
c	d	e

B.

A	*B*	*C*	*D*
a	b	c	d
b	a	d	g
d	f	g	g

C.

A	*B*	*C*
a	b	c
b	a	d

D.

A	*B*	*C*	*D*
a	b	c	d
b	a	d	g

5．关系 *R*、*S* 如下表所示，关系代数表达式$\pi_{1,5,6}(\sigma_{1>5}(R\times S))$=______。

关系 *R*

A	*B*	*C*
1	2	3
4	5	6
7	8	9
10	11	12

关系 *S*

A	*B*	*C*
3	7	11
4	7	6
5	12	13
6	10	14

A.

A	B	C
1	12	13
1	10	14

B.

A	B	C
10	7	11
10	7	6

C.

A	B	C
7	12	13
7	10	14

D.

A	B	C
4	7	6
4	7	11

6．关系 R、S 如下表所示，元组演算表达式 $\{t \mid (\forall u)(R(t) \wedge S(u) \wedge t[3] > u[1])\}$ 的结果为______。

关系 R

A	B	C
1	2	3
4	5	6
7	8	9
10	11	12

关系 S

A	B	C
3	7	11
4	5	6
5	9	13
6	10	14

A.

A	B	C
1	2	3
4	5	6

B.

A	B	C
2	7	11
5	5	6

C.

A	B	C
7	8	9
10	11	12

D.

A	B	C
5	9	13
6	10	14

7．设有关系 *R*、*S* 如下表所示，则关系代数表达式 $R \div S$ 的结果集为__(47)__。(2006 年上半年试题 47)

关系 R

A	B	C
a_1	b_1	c_1
a_1	b_2	c_1
a_2	b_2	c_2

关系 S

B	D
b_1	d_1
b_2	d_1

(47)

A.

A
a_1
a_1

B.

A
a_1

C.

A	C
a_1	c_1

D.

A	C
a_1	c_1
a_1	c_2

9.3.4 同步练习参考答案

1．(1) B (2) C

2．(1) A (2) C

3．D

4．(1) A (2) D

5．B

6．C

7．C

9.4　关系数据库 SQL 简介

9.4.1　考点辅导

9.4.1.1　SQL 数据库体系结构

SQL(Structured Query Language)是在关系数据库中最普遍使用的语言，它不仅包含数据查询功能，还包括插入、删除、更新和数据定义功能。目前，主要有 3 个标准：ANSI SQL；对 ANSI SQL 进行修改后在 1992 年采用的标准 SQL-92 或 SQL2；最近的 SQL-99 标准(也称 SQL3)。

1. SQL 的特点

SQL 具有以下特点。

(1) 综合统一。非关系模型的数据语言分为模式定义语言和数据操纵语言。其缺点是，当要修改模式时，必须停止现有数据库的运行，转储数据，修改模式并编译后再重装数据库。SQL 集数据定义语言、数据操纵语言和数据控制语言的功能于一身，语言风格统一，可独立完成数据库生命周期的所有活动。

(2) 高度非过程化。非关系数据模型的数据操纵语言是面向过程的，若要完成某项请求时，必须指定存储路径；而 SQL 是高度非过程化语言，当进行数据操作时，只要指出“做什么”，无须指出“怎么做”，存储路径对用户来说是透明的，提高了数据的独立性。

(3) 面向集合的操作方式。非关系数据模型采用的是面向记录的操作方式，操作对象是一条记录。而 SQL 采用面向集合的操作方式，其操作对象、查找结果可以是元组的集合。

(4) 两种使用方式。第一种方式是用户可以在终端键盘上输入 SQL 命令，对数据库进行操作，故称之为自含式语言；第二种方式，将 SQL 嵌入到高级语言程序中，所以又是嵌入式语言。

(5) 语言简洁，易学易用。SQL 功能极强，完成核心功能只用了 9 个动词，包括数据查询(SELECT)、数据定义(CREATE、DROP、ALTER)、数据操纵(INSERT、UPDATE、DELETE)及数据控制(GRANT、REVOKE)。

2. SQL 支持三级模式结构

SQL 支持关系数据库的三级模式结构，即视图对应外模式、基本表对应模式和存储文件对应内模式。

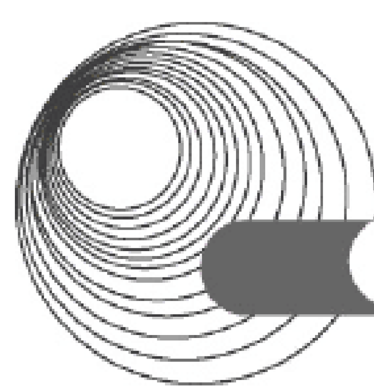
9.4.1.2　SQL 的基本组成

SQL 的基本组成如下。

(1) 数据定义语言(DDL)。SQL DDL 提供定义关系模式和视图、删除关系和视图、修改关系模式的命令。

(2) 交互式数据操纵语言(DML)。SQL DML 提供查询、插入、删除和修改的命令。

(3) 事务控制(Transaction Control)。SQL 提供的定义事务开始和结束的命令。



(4) 嵌入式 SQL 和动态 SQL。用于嵌入到某种通用的高级语言(C、C++、Java、PL/I、Cobol、VB 等)中混合编程。其中 SQL 负责操纵数据库，高级语言负责控制程序流程。

(5) 完整性(Integrity)。SQL DDL 包括定义数据库中的数据必须满足的完整性约束条件的命令，对于破坏完整性约束条件的更新将被禁止。

(6) 权限管理(Authorization)。SQL DDL 中包括说明对关系和视图的访问权限的命令。

9.4.1.3 SQL 数据定义

1. 创建表

语句格式：

```
CREATE TABLE <表名>(<列名><数据类型>[列级完整性约束条件]
                [,<列名><数据类型>[列级完整性约束条件]]…
                [,<表级完整性约束条件>]);
```

列级完整性约束条件有 NULL、UNIQUE，如 NOT NULL UNIQUE 表示取值唯一，不能取空值。

2. 修改表和删除表

1) 修改表

语句格式：

```
ALERT TABLE <表名>[ADD<新列名><数据类型>[列级完整性约束条件]]
  [DROP <完整性约束名>]
  [MODIFY <列名><数据类型>];
```

2) 删除表

语句格式：

```
DROP TABLE <表名>
```

3. 定义和删除索引

索引分聚集索引和非聚集索引。聚集索引是指索引表中索引项的顺序与表中记录的物理顺序一致的索引。

1) 建立索引

语句格式：

```
CREATE [UNIQUE] [CLUSTER] INDEX <索引名>
ON <表名>(<列名>[<次序>][,<列名>[<次序>]]…);
```

参数说明如下。

- <次序>：可选升序(ASC)或降序(DSC)，默认值为 ASC。
- UNIQUE：表明此索引的每一个索引值只对应唯一的数据记录。
- CLUSTER：表示要建立的索引是聚集索引，意为索引项的顺序与表中记录的物理顺序一致。

2) 删除索引

语句格式：

```
DROP INDEX <索引名>;
```

4. 视图创建与删除

视图是从一个或多个表或视图中导出的表，其结构和数据是建立在对表的查询基础上的。视图不是真实存在的基础表而是一个虚拟表，视图所对应的数据并不实际地以视图结构存储在数据库中，而是存储在视图所引用的表中。

1)　视图的创建

语句格式：

```
CREATE VIEW 视图名 (列表名)
AS SELECT 查询子句
[WITH CHECK OPTION];
```

注意：视图的创建中，必须遵循以下规定。

- 子查询可以是任意复杂的 SELECT 语句，但通常不允许含有 ORDER BY 子句和 DISTINCT 短语。
- WITH CHECK OPTION 表示对 UPDATE、INSERT、DELETE 操作时要保证更新、插入或删除的行满足视图定义中的谓词条件(即子查询中的条件表达式)。
- 组成视图的属性列名或者全部省略或者全部指定。如果省略属性列名，则隐含该视图由 SELECT 子查询目标列的主属性组成。

2)　视图的删除

语句格式：

```
DROP VIEW 视图名;
```

9.4.1.4　SQL 数据查询

1. SELECT 基本结构

语句格式：

```
SELECT [ALL|DISTINCT]<目标列表达式>[,<目标列表达式> ]…
FROM <表名或视图名>[,<表名或视图名>]
[WHERE<条件表达式>]
[GROUP BY <列名 1>[HAVING<条件表达式>]]
[ORDER BY<列名 2>[ASC|DESC]];
```

SQL 查询中的子句顺序：SELECT、FROM、WHERE、GROUP BY、HAVING 和 ORDER BY。SELECT、FROM 是必需的，HAVING 子句只能与 GROUP BY 搭配使用。

(1)　SELECT 子句对应的是关系代数中的投影运算，用来列出查询结果中的属性。其输出可以是列名、表达式、集函数(AVG、COUNT、MAX、MIN、SUM)，DISTINCT 选项可以保证查询的结果集中不存在重复元组。

(2)　FROM 子句对应的是关系代数中的笛卡儿积，它列出的是表达式求值过程中须扫描的关系，即在 FROM 子句中出现多个基本表或视图时，系统首先执行笛卡儿积操作。

(3)　WHERE 子句对应的是关系代数中的选择谓词。WHERE 子句的条件表达式中可以使用的运算符如表 9-3 所示。

表 9-3　WHERE 子句的条件表达式中可以使用的运算符

运算符		含义	运算符		含义
集合成员运算符	IN	在集合中	算术运算符	>	大于
	NOT IN	不在集合中		≥	大于等于
字符串匹配运算符	LIKE	与_和%进行单个多个字符匹配		<	小于
				≤	小于等于
				=	等于
				≠	不等于
空值比较运算符	IS NULL	为空	逻辑运算符	AND	与
	IS NOT NULL	不为空		OR	或
				NOT	非

2. 简单查询

SQL 最简单的查询是找出关系中满足特定条件的元组，这些查询与关系代数中的选择操作类似。简单查询只需要使用 3 个保留字，即 SELECT、FROM 和 WHERE。

3. 连接查询

若查询涉及两个以上的表，则称为连接查询。

4. 子查询与聚集函数

1) 子查询

子查询也称为嵌套查询，是指一个 SELECT-FROM-WHERE 查询可以嵌入另一个查询块之中。在 SQL 中允许多重嵌套。

2) 聚集函数

聚集函数是以一个值的集合为输入，返回单个值的函数。SQL 提供了 5 个预定义集合函数，即平均值 AVG、最小值 MIN、最大值 MAX、求和 SUM 及计数 COUNT。

使用 ANY 和 ALL 谓词必须同时使用比较运算符，其含义及等价的转换关系如表 9-4 所示。

表 9-4　ANY、ALL 谓词含义及等价的转换关系

谓词	语义	等价转换关系
>ANY	大于子查询结果中的某个值	>MIN
>ALL	大于子查询结果中的所有值	>MAX
<ANY	小于子查询结果中的某个值	<MAX
<ALL	小于子查询结果中的所有值	<MIN
>=ANY	大于等于子查询结果中的某个值	>=MIN
>=ALL	大于等于子查询结果中的所有值	>=MAX
<=ANY	小于等于子查询结果中的某个值	<=MAX

续表

谓　词	语　义	等价转换关系
<=ALL	小于等于子查询结果中的所有值	<=MIN
<>ANY	不等于子查询结果中的某个值	
<>ALL	不等于子查询结果中的任何一个值	NOT IN
=ANY	等于子查询结果中的某个值	IN
=ALL	等于子查询结果中的所有值	

5. 分组查询

1) GROUP BY 子句

在 WHERE 子句后面加上 GROUP BY 子句可以对元组进行分组，保留字 GROUP BY 后面跟着一个分组属性列表。最简单的情况是，FROM 子句后面只有一个关系，根据分组属性对其元组进行分组。SELECT 子句中使用的聚集操作符仅用在每个分组上。

2) HAVING 子句

假如元组在分组前按照某种方式加上限制，使得不需要的分组为空，则在 GROUP BY 子句后面跟一个 HAVING 子句即可。

当元组含有空值时，应注意以下两点。

- 空值在任何聚集操作中都被忽略。它对求和、求平均值和计数都没有影响，也不能是某列的最大值或最小值。
- NULL 值可以在分组属性中看作一个一般的值。

6. 更名运算

SQL 提供可为关系和属性重新命名的机制，这是通过使用具有以下形式的 AS 子句来实现的。

```
old-name AS new-name
```

AS 子句既可出现在 SELECT 子句中，也可以出现在 FROM 子句中。

7. 字符串操作

对于字符串进行的最通常的操作是使用 LIKE 操作符的模式匹配。使用两个特殊的字符来描述模式，即“%”匹配任意字符串、“-”匹配任意一个字符。

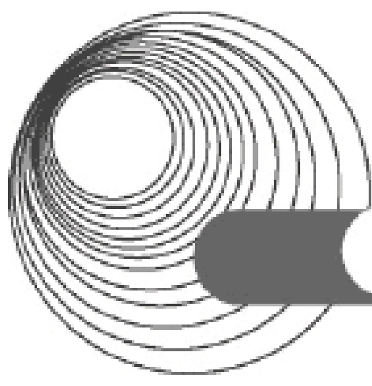
8. 视图查询

查询视图表时，系统先从数据字典中取出该视图的定义，然后将定义中的查询语句和对该视图的查询语句结合起来，形成一个修正的查询语句。

9.4.1.5　SQL 数据更新

1. 插入

要在关系数据库中插入数据，可以指定被插入的元组，或者用查询语言选出一批待插入的元组。插入语句的基本格式如下。




```
INSERT INTO 基本表名 [(字段名[,字段名>]…)]
    VALUE(常量[,常量]…);查询语句
INSERT INTO 基本表名(列表名)
    SELECT 查询语句
```

2. 删除

语句格式：

```
DELETE FROM 基本表名
    [WHERE 条件表达式]
```

3. 修改

语句格式：

```
UPDATE 基本表名
    SET 列名=值表达式[,列名=值表达式…]
    [WHERE 条件表达式];
```

9.4.1.6 SQL 的访问控制

数据控制是控制用户的数据存储权利，是由 DBA 来决定的。DBMS 数据控制应具有以下功能。

- 通过 GRANT 和 REVOKE 将授权通知系统，并存入数据词典。
- 当用户提出请求时，根据授权情况检查是否执行操作请求。

SQL 标准包括 DELETE、INSERT、SELECT 和 UPDATE 权限。

1. 授权的语句格式

授权的语句格式如下。

```
GRANT <权限>[,<权限>]…
      [ON<对象类型><对象名>]
      TO<用户>[,<用户>]…
[WITH GRANT OPTION];
```

不同类型的操作对象有不同的操作权限，常见的操作权限如表 9-5 所示。

表 9-5 常见的操作权限

对 象	对象类型	操作权限
属性列	TABLE	SELECT、INSERT、UPDATE、DELETE、ALL PRIVILEGES(4 种权限总和)
视图	TABLE	SELECT、INSERT、UPDATE、DELETE、ALL PRIVILEGES(4 种权限总和)
基本表	TABLE	SELECT、INSERT、UPDATE、DELETE、ALTER、INDEX、ALL PRIVILEGES(6 种权限总和)
数据库	DATABASE	CREATETAB 建立表的权限，可由 DBA 授予普通用户

说明如下。

(1) PUBLIC：接受权限的用户可以是单个或多个具体的用户，PUBLIC 参数可将权限

赋予全体用户。

(2) WITH GRANT OPTION：若指定了此子句，获得权限的用户还可以将权限赋予其他用户。

2. 收回权限语句格式

收回权限的语句格式如下。

```
REVOKE<权限>[,<权限>]…[ON<对象类型><对象名>]…
FROM<用户>[,<用户>]…,
```

9.4.1.7 嵌入式 SQL

SQL 提供了将 SQL 语句嵌入到某种高级语言中的使用方式，识别嵌入在高级语言中的 SQL 语句通常采用预编译的方法。该方法的关键问题是必须区分主语言中嵌入的 SQL 语句，以及主语言和 SQL 间的通信问题。采用的方法是由 DBMS 的预处理程序对源程序进行扫描，识别出 SQL 语句，把它们转换为主语言调用语句，以使主语言编译程序能识别它，最后由主语言的编译程序将整个源程序编译成目标码。

嵌入式 SQL 与主语言之间的通信采用以下 3 种方式。

(1) SQL 通信区。向主语言传递 SQL 语句执行的状态信息，使主语言能够根据此信息控制程序流程。

(2) 主变量。也称为共享变量。主语言向 SQL 语句提供参数主要通过主变量，主变量由主语言的程序定义，并用 SQL 的 DECLARE 语句说明。

(3) 游标 SQL。语言是面向集合的，一条 SQL 语句可产生或处理多条记录。而主语言是面向记录的，一组主变量一次只能放一条记录，所以引入游标，通过移动游标指针来决定获取哪一条记录。

9.4.2 典型例题分析

例 1 假设学生 Students 和教师 Teachers 的关系模式如下所示：(2013 年上半年试题 56～59)

Students(学号，姓名，性别，类别，身份证号)

Teachers(教师号，姓名，性别，身份证号)

其中，学生关系中的类别分为“本科生”和“研究生”两类。

a. 查询在读研究生的教师的平均工资，最高与最低工资的差值的 SQL 语句如下：

```
SElECT (56)
FROM Teachers , Students
WHERE (57) ;
```

b. 查询既是女教师又是研究生且工资大于等于 3500 元的身份证号和姓名的 SQL 语句如下：

```
(SELECT 身份证号,姓名
 FROM Students
 WHERE (58) )
```

```
INTERSECT (SELECT 身份证号,姓名
           FROM Teachers
           WHERE (59) )
```

(56) A. AVG(工资)AS 平均工资，MAX(工资)-MIN(工资)AS 差值
B. 平均工资 AS AVG(工资)， 差值 AS MAX(工资)-MIN(工资)
C. AVG(工资)ANY 平均工资，MAX(工资)-MIN(工资)ANY 差值
D. 平均工资 ANY AVG(工资)， 差值 ANY MAX(工资)-MIN(工资)

(57) A. Students.身份证号=Teachers.身份证号
B. Students.类别='研究生'
C. Students.身份证号=Teachers.身份证号 AND Students.类别="研究生'
D. Students.身份证号=Teachers.身份证号 OR Students.类别='研究生'

(58) A. 工资>=3500
B. 工资>='3500'
C. 性别='女'AND 类别=研究生
D. 性别='女'AND 类别='研究生'

(59) A. 工资>=3500
B. 工资>='3500'
C. 性别=女 AND 类别=研究生
D. 性别='女'AND 类别='研究生'

解析：查询在读研究生的教师的平均工资，最高与最低工资的差值的 SQL 语句为：SElECT AVG(工资)AS 平均工资，MAX(工资)-MIN(工资)AS 差值 FROM Teachers, Students WHERE Students.身份证号=Teachers.身份证号 AND Students.类别='研究生'。查询既是女教师又是研究生且工资大于等于3500元的身份证号和姓名的SQL语句：(SELECT 身份证号，姓名 FROM Students WHERE 性别='女'AND 类别='研究生')INTERSECT (SELECT 身份证号，姓名 FROM Teachers WHERE 工资>=3500)。由此可知答案。

答案：(56) A (57) C (58) D (59) A

例 2 已知关系模式：图书(图书编号，图书类型，图书名称，作者，出版社，出版日期，ISBN)，图书编号唯一识别一本图书。建立“计算机”类图书的视图 Computer-BOOK，并要求进行修改、插入操作时保证该视图只有计算机类的图书。实现上述要求的 SQL 语句如下：

```
CREATE  (55)
AS SELECT 图书编号, 图书名称, 作者, 出版社, 出版日期
FROM 图书
WHERE 图书类型='计算机'
 (56)  ;(2012 年下半年试题 55～56)
```

(55) A. TABLE Computer-BOOK　　B. VIEW Computer-BOOK
C. Computer-BOOK TABLE　　D. Computer-BOOK VIEW

(56) A. FOR ALL　　B. PUBLIC
C. WITH CHECK OPTION　　D. WITH GRANT OPTION

解析：创建视图的 SQL 语句格式如下：

```
CREATE VIEW 视图名
    AS SELECT 查询子句
    [WITH CHECK OPTION];
```

其中，WITH CHECK OPTION 表示对 UPDATE、INSERT 和 DELETE 操作时保持更新、插入或删除的行满足视图定义中的谓词条件(即子查询中的条件表达式)。

答案：(55) B　(56) C

例 3　某销售公司数据库的零件关系 P(零件号，零件名称，供应商，供应商所在地，库存量)，函数依赖集 F={零件号→零件名称，(零件号，供应商)→库存量，供应商→供应商所在地}。零件关系模式 P 属于__(54)__。

查询各种零件的平均库存量、最多库存量与最少库存量之间差值的 SQL 语句如下：

```
SELECT 零件号,零件名称,__(55)__,
FROM P
__(56)__
```

(2012 年上半年试题 54～56)

(54) A. 1NF　B. 2NF　C. 3NF　D. 4NF

(55) A. AVG(库存量)AS 平均库存量，MAX(库存量)-Min(库存量) AS 差值
　B. 平均库存量 AS AVG(库存量)，差值 AS MAX(库存量)-Min(库存量)
　C. AVG 库存量 AS 平均库存量，MAX 库存量-Min 库存量 AS 差值
　D. 平均库存量 AS AVG 库存量，差值 AS MAX 库存量-Min 库存量

(56) A. ORDER BY 供应商　B. ORDER BY 零件号
　C. GROUP BY 供应商　D. GROUP BY 零件号

解析：

1NF：关系模式 P 的每一个分量都是不可再分的数据项。

2NF：关系模式 P 属于 1NF，且每个非主属性完全依赖于码。

本题中，关系模式 P 的每个分量都不可以再分，满足 1NF 的定义。从函数依赖集 F 可以看出关系 P 的码为(零件号，供应商)，零件号决定零件名称，则零件名称不完全依赖于码，不满足 2NF 的定义。因此关系模式 P 属于 1NF。

查询平均库存量需要使用 AVG()函数。计算最大值和最小值，则需要使用 Max()和 Min()函数。SELECT 语句可以通过 AS 子句为属性重新命名，形式为 old-name AS new-name，也就是说，姓名要放在 AS 的后面。

ORDER BY 子句用于排序，GROPU BY 子句用于分组。很显然，本题要按零件进行分组。

答案：(54) A　(55) A　(56) D

例 4　将 Students 表的插入权限赋予用户 UserA，并允许其将权限授予他人，应使用的 SQL 语句如下。

GRANT__(51)__TABLE Students TO UserA__(52)__;　(2011 年下半年试题 51、52)

(51) A. UPDATE　B. UPDATE ON　C. INSERT　D. INSERT ON

(52) A. FOR ALL　B. PUBLIC
　C. WITH CHECK OPTION　D. WITH GRANT OPTION

解析：授权的语句格式如下：

```
GRANT <权限>[,<权限>]…[ON<对象类型><对象名>] TO<用户>[,<用户>]…
      [WITH GRANT OPTION];
```

权限有：SELECT、INSERT、UPDATE、DELETE，插入权限是 INSERT。使用 WITH GRANT OPTION 子句可将获得的权限赋予其他用户。

答案：(51) D　　(52) D

例 5　若有关系 $R(A, B, C, D)$和 $S(C, D, E)$，则与表达式 $\Pi_{3,4,7}(\sigma_{4<5}(R\times S))$ 等价的 SQL 语句如下。

SELECT__(53)__FROM__(54)__WHERE__(55)__;(2011 年下半年试题 53～55)

(53) A. A,B,C,D,E　　B. C,D,E
　　C. $R.A, R.B, R.C, R.D, S.E$　　D. $R.C, R.D, S.E$

(54) A. R　　B. S　　C. R,S　　D. RS

(55) A. $D<C$　　B. $R.D<S.C$　　C. $R.D<R.C$　　D. $S.D<R.C$

解析：表达式 $\Pi_{3,4,7}(\sigma_{4<5}(R\times S))$ 可分为三步进行。

第一步：求笛卡儿积 $R\times S$。$R\times S$ 共 7 列，分别为 $R.A$、$R.B$、$R.C$、$R.D$、$S.C$、$S.D$、$S.E$。

第二步：选取 $R\times S$ 中第 4 个属性值小于第 5 个属性值的元组。

第三步：对第二步的结果进行投影运算，选择第 3、4、7 列。

SELECT 语句中，SELECT 对应于关系代数中投影，第 3、4、7 列分别为 $R.C$、$R.D$、$S.E$；FROM 对应于笛卡儿积，FROM 子句中出现的多个表之间用逗号分隔；WHERE 对应于关系代数中的选择谓词，这里为 $S.D<R.C$。

答案：(53) D　　(54) C　　(55) B

例 6　某公司数据库中的元件关系模式为 P(元件号，元件名称，供应商，供应商所在地，库存量)，函数依赖集 F 如下。

F={元件号→元件名称，(元件号，供应商)→库存量，供应商→供应商所在地}

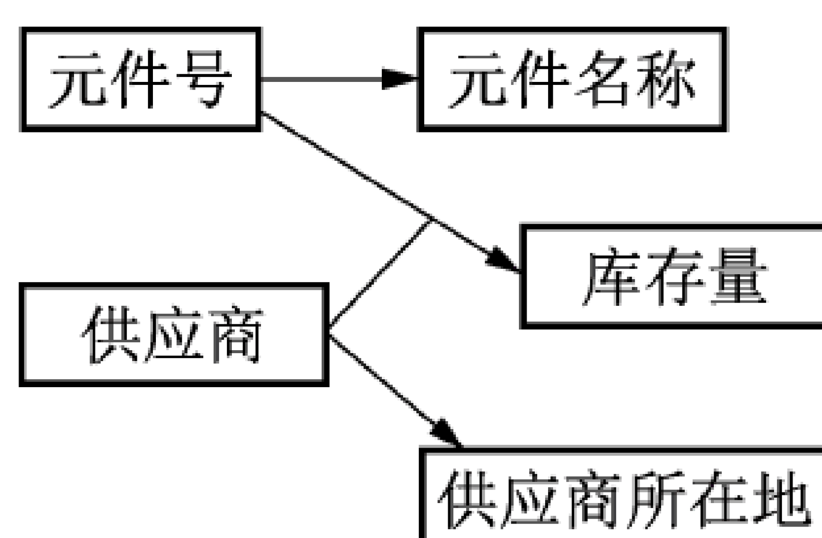
元件关系的主键为__(54)__，该关系存在冗余以及插入异常和删除异常等问题。为了解决这一问题需要将元件关系分解__(55)__，分解后的关系模式可以达到__(56)__。(2016 年上半年试题 54、55、56)

(54) A. 元件号，元件名称　　B. 元件号，供应商
　　C. 元件号，供应商所在地　　D. 供应商，供应商所在地

(55) A. 元件 1(元件号，元件名称，库存量)、元件 2(供应商，供应商所在地)
　　B. 元件 1(元件号，元件名称)、元件 2(供应商，供应商所在地，库存量)
　　C. 元件 1(元件号，元件名称)、元件 2(元件号，供应商，库存量)、元件 3(供应商，供应商所在地)
　　D. 元件 1(元件号，元件名称)、元件 2(元件号，库存量)、元件 3(供应商，供应商所在地)、元件 4(供应商所在地，库存量)

(56) A. 1NF　　B. 2NF　　C. 3NF　　D. 4NF

解析：本题第 1 空的正确选项为 B。根据题意，零件关系的主键为(零件号，供应商)。



本题第 2 空的正确选项为 C。因为关系 P 存在冗余以及插入异常和删除异常等问题。

为了解决这一问题需要将零件关系分解。选项 A、选项 B 和选项 D 是有损连接的，且不保持函数依赖性，故分解是错误的。例如，分解为选项 A、选项 B 和选项 D 后，用户无法查询某零件由哪些供应商供应，原因是分解有损连接的，且不保持函数依赖。

本题第 3 空的正确选项为 C。因为，原零件关系存在非主属性对码的部分函数依赖：(零件号，供应商)供应商所在地，但是供应商→供应商所在地，故原关系模式零件非 2NF 的。分解后的关系模式零件 1、零件 2 和零件 3 消除了非主属性对码的部分函数依赖，同时不存在传递依赖，故达到 3NF。

答案：(54) B　　(55) C　　(56) C

例 7　若给定的关系模式为 R，U={A,B,C}，F = {AB→C,C→B}，则关系 R<u>(53)</u>。(2016 年上半年试题 53)

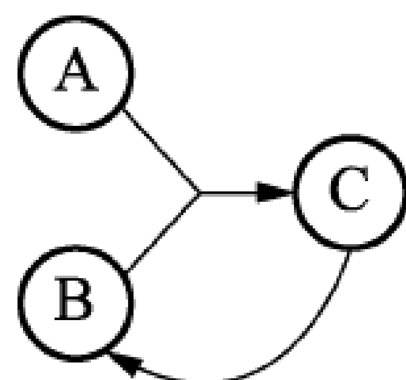
(53) A．有两个候选关键字 AC 和 BC，并且有 3 个主属性

B．有两个候选关键字 AC 和 AB，并且有 3 个主属性

C．只有一个候选关键字 AC，并且有一个非主属性和两个主属性

D．只有一个候选关键字 AB，并且有一个非主属性和两个主属性

解析：将本题关系模式 *R* 的函数依赖关系表达为下图所示。



从图中可以看出，A 的入度为零，所以它必然为候选关键字的一部分。

通过 A 与 B 组合，或 A 与 C 组合，均能遍历全图，所以候选关系字有 AB 和 AC，因此 A、B、C 均是主属性。

答案：B

9.4.3　同步练习

1．给定供应关系 SPJ(供应商号，零件号，工程号，数量)，查询某工程至少用了 3 家供应商(包含 3 家)供应零件的平均数量，并按工程号的降序排列。

```
SELECT 工程号,(1)FROM SPJ GROUP BY 工程号(2)
ORDER BY 工程号 DESC;
```

(1)　A．AVG(数量)At 平均数量

B．AVG(数量)AS 平均数量

C．平均数量 At AVG(数量)

D．平均数量 AS AVG (数量)

(2) A．HAVING COUNT(DISTINCT(供应商号))>2

B．Where COUNT(供应商号)>2

C．HAVING(DISTINCT(供应商号))>2

D．Where 供应商号 2

2．某企业职工和部门的关系模式如下，其中部门负责人也是一个职工。职工和部门关系的外键分别是 (1) 。

职工(职工号，姓名，年龄，月工资，部门号，电话，办公室)

部门(部门号，部门名，负责人代码，任职时间)

查询每个部门中月工资最高的“职工号”的 SQL 查询语句如下。

```
Select 职工号 from 职工 asE
Where 月工资=(Select Max(月工资)from 职工 asM (2))
```

(1) A．职工号和部门号　　B．部门号和负责人代码

C．职工号和负责人代码　　D．部门号和职工号

(2) A．Where M.职工号=E.职工号

B．Where M.职工号=E.负责人代码

C．Where M.部门号=部门号

D．Where M.部门号=E.部门号

9.4.4 同步练习参考答案

1．(1) B　(2) A

2．(1) B　(2) D

9.5 关系数据库规范化

9.5.1 考点辅导

9.5.1.1 函数依赖

数据依赖是通过一个关系中属性间值的相等与否体现出来的数据间的相互关系，是现实世界属性间相互联系和约束的抽象，是数据内在的性质，是语义的体现。函数依赖则是一种最重要、最基本的数据依赖。

(1) 函数依赖。设 $R(U)$是一个属性集 U 上的关系模式，X 和 Y 是 U 的子集。若对 $R(U)$的任何一个可能的关系 r，r 中不可能存在两个元组在 X 上的属性值相等，而在 Y 上的属性值不等，则称 X 函数决定 Y 或 Y 函数依赖于 X，记作 $X\rightarrow Y$。

(2) 非平凡的函数依赖。如果 $X\rightarrow Y$，但 $Y\not\subset X$，则称 $X\rightarrow Y$ 是非平凡的函数依赖。

(3) 平凡的函数依赖。如果 $X \to Y$，但 $Y \subset X$，则称 $X \to Y$ 是平凡的函数依赖。

(4) 完全函数依赖。在 $R(U)$中，如果 $X \to Y$，并且对于 X 的任何一个真子集 X'，都有 X'不能决定 Y，则称 Y 对 X 完全函数依赖，记作 $X \xrightarrow{f} Y$。

(5) 部分函数依赖。如果 $X \to Y$，但 Y 不完全函数依赖于 X，则称 Y 对 X 部分函数依赖，记作 $X \xrightarrow{p} Y$。部分函数依赖也称局部函数依赖。

(6) 传递依赖。在 $R(U, F)$中，如果 $X \to Y$，$Y \notin X$，Y 不能函数决定 X，$Y \to Z$，则称 Z 对 X 传递依赖。

(7) 码。设 K 为 $R(U, F)$中的属性的组合，若 $K \to U$，且对于 K 的任何一个真子集 K'，都有 K'不能决定 U，则 K 为 R 的候选码，若有多个候选码，则选一个作为主码。候选码通常也称为候选关键字。

(8) 主属性和非主属性。包含在任何一个候选码中的属性叫做主属性，否则叫做非主属性。

(9) 外码。若 $R(U)$中的属性或属性组 X 非 R 的码，但 X 是另一个关系的码，则称 X 为外码。

(10) 函数依赖的公理系统(Armstrong 公理系统)。设关系模式 $R(U, F)$中，U 为属性集，F 是 U 上的一组函数依赖，那么有以下的推理规则。

- A1 自反律(Reflexivity)：若 $Y \subseteq X \subseteq U$，则 $X \to Y$ 为 F 所蕴含。
- A2 增广律(Augmentation)：若 $X \to Y$ 为 F 所蕴含，且 $Z \subseteq U$，则 $XZ \to YZ$ 为 F 所蕴含。
- A3 传递律(Transitivity)：若 $X \to Y$，$Y \to Z$ 为 F 所蕴含，则 $X \to Z$ 为 F 所蕴含。

根据以上 3 条推理规则，可以推出下面 3 条推理规则。

- 合并规则：若 $X \to Y$，$X \to Z$，则 $X \to YZ$ 为 F 所蕴含。
- 伪传递率：若 $X \to Y$，$WY \to Z$，则 $XW \to Z$ 为 F 所蕴含。
- 分解规则：若 $X \to Y$ 及 $Z \subseteq Y$，则 $X \to Z$ 为 F 所蕴含。

引理　$X \to A_1 A_2 \cdots A_k$ 成立的充分必要条件是 $X \to A_i$ $(i=1, 2, \cdots, k)$成立。

9.5.1.2　规范化

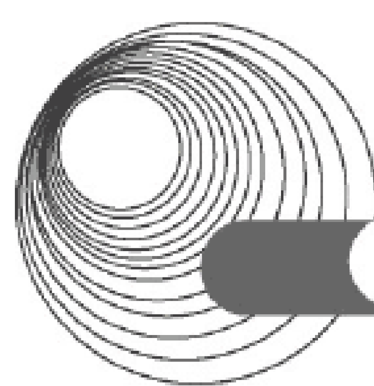
关系数据库设计的方法之一就是设计满足适当范式的模式，通常可以通过判断分解后的模式达到几范式来评价模式规范化的程度。范式有 1NF、2NF、3NF、BCNF、4NF 和 5NF，其中 1NF 级别最低。这几种范式之间 5NF ⊂ 4NF ⊂ BCNF ⊂ 3NF ⊂ 2NF ⊂ 1NF 成立。通过分解，可以将一个低一级范式的关系模式转换成若干个高一级范式的关系模式，这种过程叫做规范化。

1. 1NF(第一范式)

【定义 9-4】若关系模式 R 的每一个分量是不可再分的数据项，则关系模式 R 属于第一范式(1NF)。

1NF 存在下面 4 个问题。

- 冗余度大。
- 引起修改操作的不一致性。
- 插入异常。
- 删除异常。



2. 2NF(第二范式)

【定义 9-5】若关系模式 $R\in$1NF，且每一个非主属性完全依赖于码，则关系模式 $R\in$2NF。

换句话说，当 1NF 消除了非主属性对码的部分函数依赖，则称为 2NF。

3. 3NF(第三范式)

【定义 9-6】若关系模式 $R(U, F)$中不存在这样的码 X、属性组 Y 及非主属性 Z(Z 不属于 Y)，使得 $X\rightarrow Y$，$Y\nrightarrow X$，$Y\rightarrow Z$ 成立，则称关系模式 $R\in$3NF。

即当 2NF 消除了非主属性对码的传递函数依赖，则称为 3NF。

3NF 的模式必是 2NF 的模式。产生冗余和异常的两个重要原因是部分依赖和传递依赖。因为 3NF 模式中不存在非主属性对码的部分依赖和传递函数依赖，所以具有较好的性能。对于非 3NF 的 1NF、2NF，因其性能弱，一般不宜作为数据库模式，通常要将它们变换成为 3NF 或更高级别的范式，这种变换过程称为“关系模式的规范化处理”。

4. BCNF(巴克斯范式)

【定义 9-7】若关系模式 $R\in$1NF，若 $X\rightarrow Y$，且 Y 属于 X，X 必含有码，则关系模式 $R\in$BCNF。

即当 3NF 消除了主属性对码的部分和传递函数依赖，则称为 BCNF。

一个满足 BCNF 的关系模式，应具有以下性质。

- 所有非主属性对每一个码都是完全函数依赖。
- 所有非主属性对每一个不包含它的码，也是完全函数依赖。
- 没有任何属性完全函数依赖于非码的任何一组属性。

9.5.1.3 模式分解及分解应具有的特性

1. 分解

【定义 9-8】关系模式 $R(U, F)$的一个分解是指，$\rho=\{R_1<U_1, F_1>, R_2<U_2, F_2>, \cdots, R_n<U_n, F_n>\}$，其中 $U=U_1\cup U_2\cup\cdots\cup U_n$，并且没有 $U_i\subseteq U_j$, $1\leqslant I, j\leqslant n$，$F_i$为 F 在 U_i上的投影, $F_i=\{X\rightarrow Y\mid X\rightarrow Y\in F^+\wedge XY\subseteq U_i\}$。

对一个给定的模式进行分解，使得分解后的模式是否与原来的模式等价有 3 种情况。

- 分解具有无损连接性。
- 分解要保持函数依赖。
- 分解既要有无损连接性，又要保持函数依赖。

2. 无损连接

【定义 9-9】 $\rho=\{R_1<U_1,F_1>,R_2<U_2,F_2>,\cdots,R_n<U_n,F_n>\}$是关系模式 $R<U,F>$的一个分解，若对 R 的任何一个关系 r 均有 $r=m_\rho(r)$ 成立，则称分解 ρ 具有无损连接性(简称无损分解)。其中 $m_\rho(r)=\bowtie_{i=1}^{k}\pi_{R_i}(r)$。

【定理 9-1】关系模式 $R<U,F>$的一个分解 $\rho=\{R_1<U_1,F_1>,R_2<U_2,F_2>\}$具有无损连接的充分必要条件是

$$U_1 \cap U_2 \to U_1 - U_2 \in F^+ \text{或} U_1 \cap U_2 \to U_2 - U_1 \in F^+$$

3. 保持函数依赖

【定义 9-10】设关系模式 $R<U,F>$的一个分解 $\rho=\{R_1<U_1,F_1>,R_2<U_2,F_2>,\cdots,R_k<U_k,F_k>\}$，如果 $F^+=(\bigcup_{i=1}^{k}\pi_{R_i}(F^+))$，则称分解 ρ 保持函数依赖。

9.5.2　典型例题分析

例 1　给定关系模式 *R*(*U*，*F*)，其中：属性集 *U*={*A*,*B*,*C*,*D*,*E*,*G*}，函数依赖集 *F*={*A*→*B*，*A*→*C*，*C*→*D*，*AE*→*G*}。若将 *R* 分解为以下两个模式<u>(55)</u>，则分解后的关系模式保持函数依赖。(2013 年上半年试题 55)

(55) A. *R*1(*A*,*B*,*C*)和 *R*2(*D*,*E*,*G*)　　B. *R*1(*A*,*B*,*C*,*D*)和 *R*2(*A*,*E*,*G*)

C. *R*1(*B*,*C*,*D*)和 *R*2(*A*,*E*,*G*)　　D. *R*1(*B*,*C*,*D*,*E*)和 *R*2(*A*,*E*,*G*)

解析：对于本题来说，函数依赖集中的 *A*→*B*，*A*→*C*，*C*→*D* 在选项 B 中的 *R*1 都成立，*AE*→*G* 在 *R*2 中也成立，也就是说，每一个函数依赖左、右两边的属性都在 *R*1 或者 *R*2 中，因此得出结论，B 中的两个关系模式仍保持函数依赖。

答案：B

例 2　在数据库逻辑设计阶段，若实体中存在多值属性，那么将 E-R 图转换为关系模式时，<u>(52)</u>，得到的关系模式属于 4NF。(2015 年下半年试题 52)

(52) A. 将所有多值属性组成一个关系模式

B. 使多值属性不在关系模式中出现

C. 将实体的码分别和每个多值属性独立构成一个关系模式

D. 将多值属性和其他属性一起构成该实体对应的关系模式

解析：4NF 就是限制关系模式的属性之间不允许有非平凡且非函数依赖的多值依赖。因为根据定义，对于每一个非平凡的多值依赖 X→→Y，X 都含有候选码，于是就有 X→Y，所以 4NF 所允许的非平凡的多值依赖实际上是函数依赖。对于本题中存在多值属性的情况，应该将该实体的码和相关的多值属性独立构成一个关系模式。

答案：(52)C

例 3　设有关系模式 *R* (课程，教师，学生，成绩，时间，教室)，其中函数依赖集 *F* 如下：

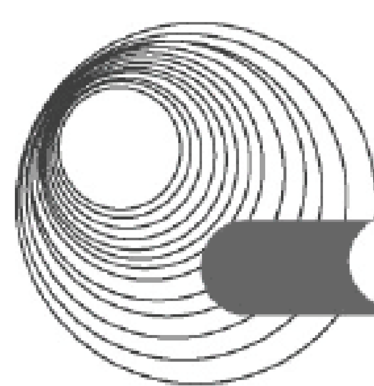
F= {课程→教师，(学生，课程)→成绩，(时间，教室)→课程，
(时间，教师)→教室，(时间，学生)→教室}

关系模式 *R* 的一个主键是<u>(54)</u>，*R* 规范化程度最高达到<u>(55)</u>。若将关系模式 *R* 分解为 3 个关系模式 *R*1(课程，教师)、*R*2(学生，课程，成绩)、*R*3(学生，时间，教室，课程)，其中 *R*2 的规范化程度最高达到<u>(56)</u>。(2010 年下半年试题 54～56)

(54) A. (学生，课程)　　B. (时间，教室)

C. (时间，教师)　　D. (时间，学生)

(55) A. 1NF　　B. 2NF　　C. 3NF　　D. BCNF



(56) A. 2NF　　B. 3NF　　C. BCNF　　D. 4NF

解析：由函数依赖关系可知，(时间，学生)是关系模式 R 的一个主键。非主属性完全依赖于主键，规范化程度可达到 2NF；2NF 消除了非主属性对主键的传递函数依赖，规范化程度为 3NF；4NF 限制关系模式的属性之间不允许有非平凡且非函数依赖的多值依赖。

答案：(54) D　　(55) B　　(56) C

例 4　假设员工关系 EMP(员工号，姓名，部门，部门电话，部门负责人，家庭住址，家庭成员，成员关系)如下表所示。如果一个部门可以有多名员工，一个员工可以有多个家庭成员，那么关系 EMP 属于(52)，且(53)问题；为了解决这一问题，应该将员工关系 EMP 分解为(54)。(2009 年上半年试题 52～54)

员工号	姓名	部门	部门电话	部门负责人	家庭住址	家庭成员	成员关系
0011	张晓明	开发部	808356	0012	北京海淀区 1 号	张大军	父亲
0011	张晓明	开发部	808356	0012	北京海淀区 1 号	胡敏铮	母亲
0011	张晓明	开发部	808356	0012	北京海淀区 1 号	张晓丽	妹妹
0012	吴俊	开发部	808356	0012	上海昆明路 15 号	吴胜利	父亲
0012	吴俊	开发部	808356	0012	上海昆明路 15 号	王若垚	母亲
0021	李立丽	市场部	808358	0021	西安雁塔路 8 号	李国庆	父亲
0021	李立丽	市场部	808358	0021	西安雁塔路 8 号	罗明	母亲
0022	王学强	市场部	808356	0021	西安太白路 2 号	王国钧	父亲
0031	吴俊	财务部	808360		西安科技路 18 号	吴鸿翔	父亲

(52) A. 1NF　　B. 2NF　　C. 3NF　　D. BCNF

(53) A. 无冗余、无插入异常和删除异常
　　B. 无冗余，但存在插入异常和删除异常
　　C. 存在冗余，但不存在修改操作的不一致
　　D. 存在冗余、修改操作的不一致，以及插入异常和删除异常

(54) A. EMP1(员工号，姓名，家庭住址)
　　　EMP2(部门，部门电话，部门负责人)
　　　EMP3(员工号，家庭成员，成员关系)
　　B. EMP1(员工号，姓名，部门，家庭住址)
　　　EMP2(部门，部门电话，部门负责人)
　　　EMP3(员工号，家庭成员，成员关系)
　　C. EMP1(员工号，姓名，家庭住址)
　　　EMP2(部门，部门电话，部门负责人，家庭成员，成员关系)
　　D. EMP1(员工号，姓名，部门，部门电话，部门负责人，家庭住址)
　　　EMP2(员工号，家庭住址，家庭成员，成员关系)

解析：BC 范式的定义如下。

关系模式 R 满足 BC 范式，当且仅当若非平凡函数依赖 $A1A2\cdots An \rightarrow B1B2\cdots Bm$ 在关系 R 中成立，则 $\{A1,A2,\cdots,An\}$ 是 R 的超键。

关系 R 满足 BC 范式的两种情况如下。

- 关系 R 中不存在非平凡函数依赖(只有平凡函数依赖)。
- 每个非平凡函数依赖的左面包含某个键(即左面是超键)。

第三范式的定义：关系模式 R 满足 3NF，当且仅当若非平凡函数依赖 $A1,A2,\cdots,An \rightarrow B$ 在关系 R 中成立，则 $\{A1,A2,\cdots,An\}$ 是 R 的超键，或者 B 是某个键的组成部分(键属性)。

第一范式的条件：每个元组的每个分量都是原子的。

第二范式的条件：在 1NF 的基础上，要求每个非键属性依赖于键的整体(直接或间接)，而不是键的部分属性，即不允许有非平凡函数依赖的右面是非键属性，而左面是某个键的真子集。(允许传递依赖存在)

满足 2NF 的几种情形如下。

- 不存在非平凡函数依赖。
- 若存在非平凡函数依赖，则其右面是某个键的组成部分(键属性)。
- 若存在非平凡函数依赖，且其右面是非键属性，则其左面要么是超键，要么包含非键属性。

对于员工关系 EMP(员工号，姓名，部门，部门电话，部门负责人，家庭住址，家庭成员，成员关系)有下列函数依赖：员工号→姓名，部门，部门电话，部门负责人，家庭住址。员工号，成员关系→家庭成员。部门→部门电话，部门负责人。由此可见主键为{员工号，成员关系}。

第一个函数依赖：员工号→姓名，部门，部门电话，部门负责人，家庭住址。

右边为非键属性，而员工号为键的组成部分。不符合 2NF，故是 1NF。

关系模式设计中可能出现各种冗余，即同一事实在多个元组中重复。造成冗余的原因通常是将同一个对象的单值和多值特征混合在同一个关系中。例如，表中的员工号为 0011 的 3 个元组存在冗余信息。

修改异常：修改某个元组的信息，而重复的信息可能未修改而破坏一致性。或插入数据时，某些有用信息暂时无法插入。

删除异常：删除某个对象时，必须删除多个元组而不是一个元组，操作不当有可能破坏数据一致性。或删除元组时，同时删除了其他有用信息。例如，删除员工号 0011 对象时，必须删除 3 个元组。

为了避免异常，用几个关系代替原有的关系，且保持数据一致性。从而进行关系的分解。可将函数依赖{员工号→姓名，部门，部门电话，部门负责人，家庭住址}分解为 EMP1(员工号，姓名，部门，家庭住址)和 EMP2(部门，部门电话，部门负责人)；函数依赖{员工号，成员关系→家庭成员}形成 EMP3(员工号，家庭成员，成员关系)。3 个关系的主键依次为员工号；部门；员工号和成员关系。

答案：(52) A　　(53) D　　(54) B

9.5.3　同步练习

1.　建立一个供应商、零件数据库。其中“供应商”表 S(Sno，Sname，Zip，City)分别表示供应商代码、供应商名、供应商邮编、供应商所在城市，其函数依赖为 Sno→(Sname，

Zip，City)，Zip→City。“供应商”表 S 属于____。

A．1NF　　B．2NF　　C．3NF　　D．BCNF

2．设关系模式 $R(A, B, C)$，传递依赖指的是 (1) ；下列结论错误的是 (2) 。

(1) A．若 $A \to B$，$B \to C$，则 $A \to C$　B．若 $A \to B$，$A \to C$，则 $A \to BC$

C．若 $A \to C$，则 $AB \to C$　D．若 $A \to BC$，则 $A \to B$，$A \to C$

(2) A．若 $A \to BC$，则 $A \to B$，$A \to C$

B．若 $A \to B$，$A \to C$，则 $A \to BC$

C．若 $A \to C$，则 $AB \to C$

D．若 $AB \to C$，则 $A \to C$，$B \to C$

3．给定关系 $R(A1，A2，A3，A4)$上的函数依赖集 $F=\{A1A3 \to A2，A2 \to A3\}$，$R$ 候选关键字为 (1) 。分解$\rho=\{(A1，A2)，(A1，A3)\}$ (2) 。

(1) A．$A1$、$A3$　B．$A1$、$A2$、$A3$　C．$A1$、$A3$、$A4$　D．$A2$ 和 $A1$、$A3$

(2) A．是无损连接的　B．是保持函数依赖的

C．既是无损连接又保持函数依赖　D．既是有损连接又不保持函数依赖

4．关系模式 $R(U，F)$，其中 $U=(W，X，Y，Z)$，$F=\{WX \to Y，W \to X，X \to Z，Y \to W\}$。关系模式 R 的候选码是 (1) ， (2) 是无损连接并保持函数依赖的分解。

(1) A．W 和 Y　B．WY　C．WX　D．WZ

(2) A．$\rho=\{R1(WY)，R2(XZ)\}$　B．$\rho=\{R1(WZ)，R2(XY)\}$

C．$\rho=\{R1(WXY)，R2(XZ)\}$　D．$\rho=\{R1(WX)，R2(YZ)\}$

5．关系规范化在数据库设计的 (52) 阶段进行。(2016 年上半年试题 52)

(52) A．需求分析　B．概念设计　C．逻辑设计　D．物理设计

9.5.4　同步练习参考答案

1．B　　2．(1) A　(2) D

3．(1) A　(2) D　　4．(1) A　(2) C

5．C

9.6　数据库的控制功能

9.6.1　考点辅导

9.6.1.1　事务管理

事务是一个操作序列，是数据库环境中不可分割的逻辑工作单位。

事务的 4 个特性是原子性、一致性、隔离性和持久性。

- 原子性。事务的所有操作在数据库中要么全做，要么全都不做。
- 一致性。一个事务独立执行的结果，将保持数据的一致性，即数据不会因为事务

的执行而被破坏。

- 隔离性。一个事务的执行不能被其他事务干扰。
- 持久性。一个事务一旦提交，它对数据库中数据的改变必须是永久的，即便系统出现故障时也是如此。

9.6.1.2　数据库的备份与恢复

1. 故障类型

人为错误、硬盘损坏、计算机病毒、断电或是天灾人祸等都有可能造成数据的丢失，所以应该强调备份的重要性。备份实际上是数据的保护意识，在危机四伏的网络环境中，数据随时有被毁灭的可能。在数据库中的四类故障有事务内部故障、系统故障、介质故障和计算机病毒。

2. 备份方法

恢复的基本原理是“建立数据冗余”(重复存储)。建立冗余数据的方法是进行数据转储和登记日志文件。数据的转储分为静态转储和动态转储、海量转储和增量转储。

(1) 静态转储和动态转储。静态转储是指在转储期间不允许对数据库进行任何存取、修改操作；动态转储是指在转储期间允许对数据库进行存取、修改操作。因此，转储和用户事务可并发执行。

(2) 海量转储和增量转储。海量转储是指每次转储全部数据；增量转储是指每次只转储上次转储后更新过的数据。

(3) 日志文件。在事务处理的过程中，DBMS 把事务开始、事务结束以及对数据库的插入、删除和修改的每一次操作写入日志文件。一旦发生故障，DBMS 的恢复子系统利用日志文件撤销事务对数据库的改变，回退到事务的初始状态。因此，DBMS 利用日志文件来进行事务故障恢复和系统故障恢复，并可协助后备副本进行介质故障恢复。

3. 恢复

数据恢复有 3 个步骤。

(1) 反向扫描文件日志，查找该事务的更新操作。

(2) 对事务的更新操作执行逆操作。

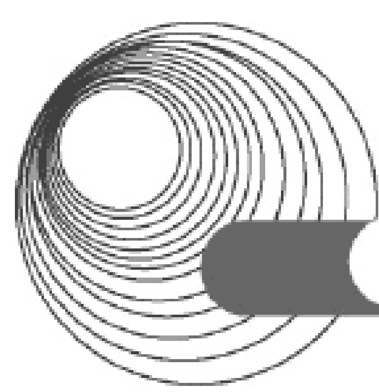
(3) 继续反向扫描日志文件，查找该事务的其他更新操作，并做同样的处理，直到事务的开始标志。

4. 数据库镜像

为了避免磁盘介质出现故障影响数据库的可用性，许多 DBMS 提供数据镜像功能用于数据库恢复。数据库镜像是通过复制数据实现的，但频繁地复制数据会降低系统的运行效果，因此实际应用中往往对关键的数据和日志文件镜像。

9.6.1.3　并发控制

并发操作是指在多用户共享的系统中，许多用户可能同时对同一数据进行操作。并发操作带来问题的原因是事务的并发操作破坏了事务的隔离性。DBMS 的并发控制子系统负责协调并发事务的执行，保证数据库的完整性不被破坏，避免用户得到不正确的数据。



1. 并发操作带来的问题

并发操作带来的数据不一致性有 3 类，即丢失修改、不可重复读和读“脏”数据。

2. 并发控制技术

并发控制的主要技术是封锁。

1) 封锁

(1) 排他锁(X 锁)。若事务 T 对数据对象 A 加上 X 锁，则只允许 T 读取和修改 A，其他事务都不能再对 A 加任何类型的锁，直到 T 释放 A 上的锁。

(2) 共享锁(S 锁)。若事务 T 对数据对象 A 加上 S 锁，则只允许 T 读取 A，但不能修改 A，其他事务只能再对 A 加 S 锁，直到 T 释放 A 上的 S 锁。

2) 三级封锁协议

(1) 一级封锁协议。事务在修改数据 R 之前必须先对其加 X 锁，直到事务结束才释放。一级封锁协议可以解决丢失更新问题。

(2) 二级封锁协议。在一级封锁协议的基础上，加上事务 T 在读取数据 R 前必须先对其加 S 锁，读完后即可释放 S 锁。二级封锁协议可以解决读“脏”数据的问题，但是由于二级封锁协议读完数据后即可释放 S 锁，所以它不能保证可重复读。

(3) 三级封锁协议。在一级封锁协议的基础上，加上事务 T 在读取数据 R 之前必须先对其加 S 锁，直到事务结束才释放。三级封锁协议可以防止丢失修改、读“脏”数据和不可重复读。

3. 活锁和死锁

活锁是指当事务 T1 封锁了数据 R，事务 T2 请求封锁数据 R，于是 T2 等待。T3 也请求封锁 R，当 T1 释放 R 上的封锁后，系统首先批准 T3 的请求，于是 T2 仍等待。然后 T4 又请求封锁 R，当 T3 释放 R 上的封锁之后系统首先批准 T4 的请求……T2 可能永远等待。

死锁是指两个以上的事务分别请求封锁对方已经封锁的数据，导致长期等待而无法继续运行下去的现象。

4. 并发调度的可串行性

【定义 9-11】多个事务的并行执行是正确的，当且仅当其结果与某一次序串行地执行它们时的结果相同时，这种调度策略称为可串行化的调度。

可串行性是并行事务正确性的准则，按照这个准则规定，一个给定的并发调度，当且仅当它是可串行化的才认为是正确调度。

5. 两段封锁协议

两段封锁协议是指所有事务必须分两个阶段对数据加锁和解锁：第一阶段是获得封锁；第二阶段是释放封锁。

6. 封锁的粒度

封锁对象的大小称为封锁的粒度。封锁的对象可以是逻辑单元也可以是物理单元。

9.6.2　典型例题分析

例 1　若事务 T 对数据 A 已加排他锁，那么其他事务对数据 A(56)。(2007 年下半年试题 56)

(56) A．加共享锁成功，加排他锁失败　　B．加排他锁成功，加共享锁失败
C．加共享锁、加排他锁都成功　　D．加共享锁、加排他锁都失败

解析：本题主要考查排他锁与共享锁的区别。

排他锁又称写锁(eXclusive Lock，X 锁)，如果一个事务 T 对数据 A 加排他锁，则事务 T 只能读取和修改 A，其他事务不能再对 A 加任何类型的锁，直到 T 释放为止。其作用主要就是为了避免其他事务获取资源上的锁。一般在事务的更新操作过程中始终应用排他锁。

共享锁又称读锁(Share Lock，S 锁)，如果事务 T 对数据 A 加共享锁，那么其他事务也只能对 A 加 S 锁，不能加其他锁，直到 T 释放 S 锁为止。

答案：D

例 2　操作序列 T1、T2、T3 对数据 A、B、C 并发操作如下表所示，T1 与 T2 间并发操作 (50) ，T2 与 T3 间并发操作 (51) 。(2006 年下半年试题 50、51)

时　间	T1	T2	T3
t1			
t2	读 A=50		
t3	读 B=200		
t4	X1=A+B		
t5			读 B=200
t6			
t7		读 B=200	
t8		B=B-100	
t9	读 A=50	写 B	
t10	读 B=200		
t11	X1=A+B		
t12	验算不对		B=B+50 写 B

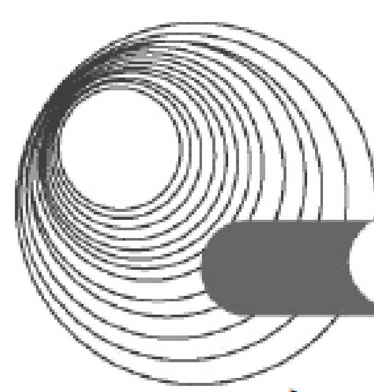
(50)～(51) A．不存在问题　　B．将丢失修改
C．不能重复读　　D．将读“脏”数据

解析：此题主要考查数据库的并发控制。

并发操作带来的数据不一致有 3 类，即丢失修改、不可重复读和读“脏”数据。首先看一下这 3 个概念。

丢失修改：当多个事务选择同一行数据，然后基于最初选定的值更新该值时，就会发生丢失更新问题。因为每个事务都无法感知其他事务的存在，最后的更新将重写其他事务所做的更新，由此会带来数据的丢失问题。

不可重复读：在一个事务内，多次读同一数据。在时间交叠的范围内，另外一个事务



也访问同一数据。这样，在第一个事务两次读数据之间，由于第二个事务可能对数据进行了修改，而造成读出的数据不一致的问题，因此称为不可重复读。

读“脏”数据：指一个事务访问数据时对其进行了修改，但是还没来得及提交数据库，而此时另外一个事务也访问此数据。这时，因为数据还没有来得及提交数据库，那么第二个事务读到的是旧的数据，称为脏数据。

看本题目，T1 两次读同一数据 A、B，而这两次之间 T2 对数据 B 进行了修改，造成 T1 的不可重复读。

而 T2、T3 读取同一数据 B，且对其进行了更新，但是因为 T3 操作在 T2 之后，覆盖 T2 的操作，造成更新丢失。因此，可以得出本题答案为 C、B。

答案：(50) C (51) B

例 3 计算机系统的软、硬件故障可能会造成数据库中的数据被破坏。为了防止这一问题，通常需要 (56) ，以便发生故障时恢复数据库。(2013 年下半年试题 56)

(56) A．定期安装 DBMS 和应用程序
B．定期安装应用程序，并将数据库做镜像
C．定期安装 DBMS，并将数据库作备份
D．定期将数据库作备份；在进行事务处理时，需要将数据更新写入日志文件

解析：为了防止计算机系统的软、硬件故障可能会造成数据库中的数据被破坏，通常采用定期将数据库作备份；在进行事务处理时，需要将数据更新写入日志文件。答案选 A。

答案：D

9.6.3 同步练习

1. “要么都做，要么都不做”是对事物 4 个特性之一_____的描述。
A．一致性 B．原子性 C．隔离性 D．持久性

2. 当事务 T1 封锁了数据 R，事务 T2 请求封锁数据 R，于是 T2 等待。T3 也请求封锁 R，当 T1 释放 R 上的封锁后，系统首先批准 T3 的请求，于是 T2 仍等待。然后 T4 又请求封锁 R，当 T3 释放 R 上的封锁之后系统首先批准了 T4 的请求……T2 可能永远等待，这种现象描述的是_____。
A．排他锁 B．共享锁 C．活锁 D．死锁

9.6.4 同步练习参考答案

1．B 2．C

9.7 本 章 小 结

本章知识点在 2013 年的新大纲中改动不大，只是有一些描述方面的调整。

本章主要要求考生掌握数据库的基本概念和特征；了解数据库管理系统的功能和特征，

理解数据库模型，包括概念模式、外模式和内模式；掌握 E-R 图，理解第一、二、三范式的定义；会基本的数据操作，包括集合运算和关系运算；掌握 SQL 语言；了解数据库的控制功能，包括并发控制、恢复、安全性和完整性。关于考试所占比例，在每次考试的上午试题中大概考 5～6 道题。

数据库的知识点较多，且大多比较抽象、晦涩。这就要求考生在复习时，不能死记硬背，应首先理解，在理解的基础上加强练习。同时还要注意这些知识点之间的联系，建立整体观念，做到融会贯通。比如 SQL 的功能就十分丰富。

9.8　达标训练题及参考答案

9.8.1　达标训练题

1．设关系 P 和 Q 分别为二元和三元关系，则与关系代数表达式 $P \underset{1<2}{\bowtie} Q$ 等价的是____。

A．$\sigma_{1<2}(P\times Q)$　　B．$\sigma_{1<4}(P\times Q)$　　C．$\sigma_{1<2}(P\bowtie Q)$　　D．$\sigma_{1<4}(P\bowtie Q)$

2．最常用的一种基本数据模型是关系数据模型，它用统一的(1)结构来表示实体及实体之间的联系。关系数据库的数据操作语言(DML)主要包括(2)两类操作。

关系运算以关系代数为理论基础，关系代数的最基本操作是并、差、笛卡儿积、(3)。用 $R|\times|S$ 表示关系 R 和关系 S 的(4)。

设关系 R 和关系 S 如下表所示。

	A	B	C		B	C	D		A	B	C	D
R	a	b	c	S	b	c	d	T	a	b	c	d
	b	b	f		b	c	e		a	b	c	e
	c	a	d		a	d	b		c	a	d	b
	d	a	d						d	a	d	b

则关系 T 是关系 R 和关系 S(5)的结果。

(1) A．树　B．网络　C．图　D．二维表

(2) A．插入和删除　B．检索和更新　C．查询和编辑　D．统计和修改

(3) A．投影、连接　B．连接、选择　C．选择、投影　D．交、选择

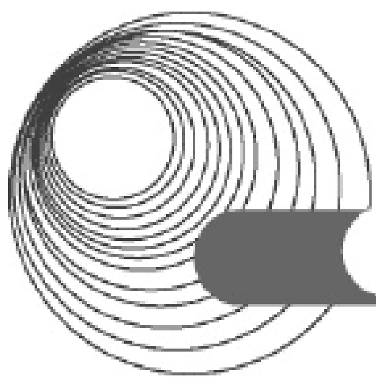
(4) A．联接　B．笛卡儿积　C．θ 连接　D．自然连接

(5) A．自然连接　B．θ 连接　C．笛卡儿积　D．并

3．域表达式$\{ab|R(ab)\wedge R(ba)\}$转换成为等价的关系代数表达式，所列出的式子中(1)是不正确的。SQL 中集合成员资格的比较操作“元组 NOT IN(集合)”中的 NOT IN 与(2)操作符等价。SQL 中涉及属性 AGE 是否是空值的比较操作，写法(3)是错误的。类似于“工资在 800～5000 之间”这种约束，是属于 DBS 的(4)功能。

设关系模式 Q 是 3NF 模式，那么，(5)这种提法是不正确的。

(1) A．$\pi_{1,2}(\sigma_{1=4\wedge 2=3}(R\times R))$　　B．$\pi_{1,2}(\sigma_{1=4}(R \underset{2=3}{\bowtie} R))$



C．$\pi_{1,2}(\sigma_{1=4}(R \underset{2=1}{\bowtie} R))$　　D．$R \cap \pi_{2,1}(R)$

(2) A．<>SOME　B．=SOME　C．<>ALL　D．=ALL

(3) A．AGE IS NULL　B．NOT(AGE IN NULL)

C．AGE=NULL　D．AGE IS NOT NULL

(4) A．完整性　B．并发控制　C．安全性　D 恢复

(5) A．Q 一定是 2NF 模式　B．Q 可能不是 4NF 模式

C．Q 可能不是 BCNF　D．Q 一定不是 BCNF

9.8.2 参考答案

1．B

2．(1) B　(2) D　(3) D　(4) C　(5) B

3．(1) B　(2) C　(3) C　(4) D　(5) A

第 10 章　网络与信息安全基础知识

大纲要求：

- 网络体系结构(网络拓扑、OSI/RM、基本的网络协议)。
- 传输介质、传输技术、传输方法、传输控制。
- 常用网络设备和各类通信设备。
- 客户机/服务器结构、浏览器/服务器结构。
- LAN 的拓扑、存取控制、LAN 的组网、LAN 间连接、LAN-WAN 连接。
- 因特网基础知识及应用。
- 网络软件。
- 网络管理。
- 网络性能分析。
- 安全性基本概念。
- 防治计算机病毒，防范计算机犯罪。
- 存取控制、防闯入、安全管理措施。

10.1　网 络 概 述

近 4 年没有考查本节内容，不过建议考生应掌握以下内容。

1. 计算机网络的概念

1)　计算机网络的发展

计算机网络是现代通信技术与计算机技术相结合的产物，它的发展过程大致可以划分为 4 个阶段。

(1)　具有通信功能的单机系统。

(2)　具有通信功能的多机系统。

(3)　以共享资源为目的的计算机网络。

(4)　以局域网及互联网为支撑的分布式计算机系统。

2)　计算机网络的功能

计算机网络的主要功能有数据通信、资源共享、负载均衡和高可靠性。

2. 计算机网络的分类

计算机网络的分类方式很多，按照不同的分类原则，可以得到各种不同类型的计算机网络。例如，按通信距离可分为广域网、局域网和城域网；按信息交换方式可分为电路交换网、分组交换网和综合交换网；按网络拓扑结构可分为星型网、树型网、环型网和总线网；按通信介质可分为双绞线网、同轴电缆网、光纤网和卫星网等；按传输带宽可分为基

带网和宽带网；按使用范围可分为公用网和专用网；按速率可分为高速网、中速网和低速网；按通信传播方式可分为广播式和点到点式。

下面介绍按通信距离对计算机网络的分类。

(1) 局域网(LAN)。局域网是指传输距离有限，传输速率较高，以共享网络资源为目的的网络系统。其特点如下：①分布范围有限，加入局域网中的计算机通常处在几千米以内；②有较高的通信带宽，数据传输率高；③数据传输可靠，误码率低；④通常采用同轴电缆或双绞线作为传输介质，跨楼栋时使用光纤；⑤拓扑结构简单；⑥网络的控制一般趋向于分布式；⑦通常网络归单一组织所拥有和使用。

(2) 城域网(MAN)。城域网是规模介于局域网和广域网之间的一种较大范围的高速网络，一般覆盖邻近的多个单位和城市，从而为接入网络的企业、机关、公司及社会单位提供文字、声音和图像的集成服务。

(3) 广域网(WAN)。又称为远程网。它是覆盖范围广、传输速率相对较低、以数据通信为主要目的的数据通信网。其特点如下：①分布范围广；②数据传输率低；③数据传输可靠性随着传输介质的不同而不同；④广域网常常借用传统的公共传输网来实现；⑤拓扑结构较为复杂。

3. 网络的拓扑结构

1) 总线结构

总线结构的特点如下：①总线拓扑结构中只有一条双向通路，便于进行广播式传送信息；②总线拓扑结构属于分布式控制；③节点的增删和位置的变动较容易；④节点的接口通常采用无源电路；⑤设备少，价格低，安装使用方便；⑥对信号的质量要求高等。

2) 星状结构

星状结构中，使用中央交换处理单元以放射状连接到网中的各个节点。中央单元采用电路交换方式以建立所希望通信的两节点间专用的路径，通常用双绞线将节点与中央单元进行连接。其特点是：①维护管理容易；②故障隔离和检测容易；③网络延迟时间短等。

3) 环状结构

环状结构的信息传输线路构成一个封闭的环，各节点通过中继器连入网内，各中继器间首尾相接。信息单向沿环路逐点传送。其特点是：①环状网中信息的流动方向是固定的，两个节点仅有一条通路；②有旁路设备；③信息要串行穿过多个节点，系统响应速度慢等。

4) 树状结构

树状结构是总线结构的扩充形式，传输介质是不封闭的分支电缆，主要用于多个网络组成的分级结构中。其特点同总线结构。

5) 分布式结构

分布式结构无严格的布点规定和形状，各节点之间有多条线路相连。其特点是：有较高的可靠性；资源共享方便，网络响应时间短；节点的路由选择和流量控制难度大，管理软件复杂；硬件成本高。

10.2　ISO/OSI 网络体系结构

10.2.1　考点辅导

国际标准化组织(ISO)提出了开放系统互连参考模型(OSI)，它是一个定义异种计算机连接标准的框架结构，共有 7 层。

(1) 物理层。OSI 的第一层，提供为建立、维护和拆除物理链路所需的机械、电气、功能和规程的特性；提供有关在传输介质上传输非结构的位流及物理链路故障检测指示。

(2) 数据链路层。负责在两个相邻节点间的线路上无差错地传送以帧为单位的数据，并进行流量控制。

(3) 网络层。为传输层实体提供端到端的交换网络数据传送功能，使得传输层摆脱路由选择、交换方式、拥塞控制等网络传输细节；可以为传输层实体建立、维持和拆除一条或多条通信路径。对网络传输中发生的不可恢复的差错予以报错。

(4) 传输层。为会话层实体提供透明、可靠的数据传输服务，保证端到端的数据完整性；选择网络层能提供最适宜的服务；提供建立、维护和拆除传输连接功能。

(5) 会话层。为彼此合作的表示层实体提供建立、维护和结束会话连接的功能；完成通信进程的逻辑名字与物理名字间的对应；提供会话管理服务。

(6) 表示层。为应用层进程提供解释所交换信息含义的一组服务。数据的压缩、解压缩、加密和解密等工作都由表示层负责。

(7) 应用层。提供 OSI 用户服务，即确定进程之间通信的性质，以满足用户需要以及提供网络与用户应用软件之间的接口服务，包括事务处理程序、电子邮件和网络管理程序等。

10.2.2　典型例题分析

例 1　以下关于层次化局域网模型中核心层的叙述，正确的是 (70) 。(2015 年下半年试题 70)

(70) A．为了保障安全性，对分组要进行有效性检查
　　B．分组从一个区域高速地转发到另一个区域
　　C．由多台二、三层交换机组成
　　D．提供多条路径来缓解通信瓶颈

解析：层次化模型中最为经典的是 3 层模型，主要将网络划分为核心层、汇聚层和介入层。核心层一般由经过可用性和性能优化的高端路由器和交换机组成，提供不同区域或者下层的高速连接和最优传送路径；汇聚层由用于实现策略的路由器或者交换机构成，将网络业务连接到接入层，并且实施与安全、流量负载和路由相关的策略；接入层由连接用户的低端交换机构成，为局域网接入广域网或者终端用户访问网络提供接入服务。

答案：(70) B

例 2　以下关于 VLAN 的叙述中，属于其优点的是__(47)__。(2015 年上半年试题 47)

(47) A．允许逻辑地划分网段　　B．减少了冲突域的数量
　　C．增加了冲突域的大小　　D．减少了广播域的数量

解析： VLAN 即虚拟局域网，它根据管理功能、组织机构和应用类型对交换局域网进行分段而形成逻辑网络。属于同一个 VLAN 的所有端口构成一个广播域。把物理网络划分为 VLAN，可以控制网络流量，有助于控制广播风暴，减小冲突域，提高网络利用率。

答案： A

10.2.3　同步练习

1．从供选择的答案中，选出应填入下面叙述中_____内的最确切的解答，把相应编号写在答卷的对应栏内。

1)　电子商务(1)　　2)　人机接口(2)

3)　计算机性能评价(3)　　4)　并行处理(4)

5)　网络体系结构(5)

(1)～(5) A．OSI、对等层协议、无连接服务　B．OSI、对等层协议、TPC
　C．OCR、基准测试程序、TPC　D．吉普森方接、基准测试程序、TPC
　E．EDI、网上商店、身份验证　F．向量计算机、MPP、多指令流
　G．EDI、网上商店、OCR　H．所见即所得、语音识别、OCR
　I．EDI、MPP、多指令流　J．语音识别、OCR、OSI

2．　在 OSI 参考模型中，数据链路层处理的数据单位是_____。

A．比特　　B．帧　　C．分组　　D．报文

10.2.4　同步练习参考答案

1．(1) E　(2) H　(3) D　(4) F　(5) I

2．B

10.3　网络互联的硬件

10.3.1　考点辅导

10.3.1.1　网络的设备

1．网络传输介质互联设备

网络传输介质互联设备包括 T 形头、收发器、屏蔽或非屏蔽双绞线连接器 RJ-45、RS-232 接口、DB-15 接口、VB35 同步接口、网络接口单元、调制解调器等。

2. 物理层互联设备

物理层互联设备包括中继器和集线器。

(1) 中继器。由于信号在网络传输介质中有衰减和噪声，使有用的数据信号变得越来越弱，因此为了保证有用数据的完整性，并在一定范围内传送，要用中继器把所接收到的弱信号分离，并再生放大以保持与原数据相同。其主要优点是安装简单、使用方便、价格便宜。

(2) 集线器。集线器可以说是一种特殊的多路中继器，也有信号放大的功能。使用双绞线的以太网多用集线器扩大网络，同时便于网络的维护。以集线器为中心的优点是：当网络系统中某条线路或某节点出现故障时，不会影响网上其他节点的正常工作。集线器分为无源集线器、有源集线器和智能集线器。

3. 数据链路层互联设备

数据链路层互联设备包括网桥和交换机。

(1) 网桥。网桥是一个局域网与另一个局域网之间建立连接的桥梁，它的作用是扩展网络和通信手段，在各种传输介质中转发数据信号，扩展网络的距离，同时又有选择地将有地址的信号从一个传输介质发送到另一个传输介质，并能有效地限制两个介质系统中无关紧要的通信。

(2) 交换机。交换机是一个具有简化、低价、高性能和高端口密集特点的交换产品。交换技术允许共享型和专用型的局域网段进行带宽调整，以减轻局域网之间信息流通出现的瓶颈问题。交换机的 3 种交换技术为端口交换、帧交换和信元交换。

4. 网络层互联设备

网络层互联设备是路由器，其用于连接多个逻辑上分开的网络；具有很强的网络互联能力；具有判断网络地址和选择路径的功能。缺点是由于工作在网络层，处理的信息比网桥要多，因而处理速度比网桥慢。

5. 应用层互联设备

应用层互联设备是网关，其体现在 OSI 模型的最高层，它将协议进行转换，将数据重新分组，以便在两个不同类型的网络系统之间进行通信。

10.3.1.2 网络的传输介质

1. 有线介质

有线介质包括以下几种。

(1) 双绞线。

(2) 同轴电缆。

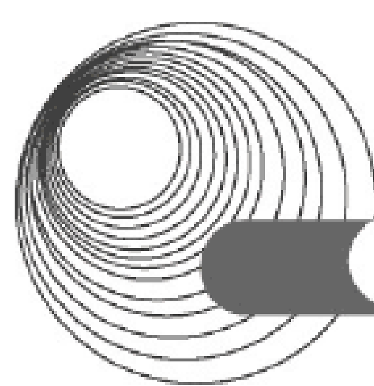
(3) 光纤。

2. 无线介质

无线介质包括以下几种。

(1) 微波。

(2) 红外线和激光。



(3) 卫星通信。

10.3.1.3 组建网络

在一个局域网中，其基本组成部件为服务器、客户机、网络设备、通信介质、网络软件等。

(1) 服务器。它是局域网的核心，可进一步分为文件服务器、打印服务器和通信服务器。

(2) 客户机。客户机又称为用户工作站，是用户与网络应用接口设备。

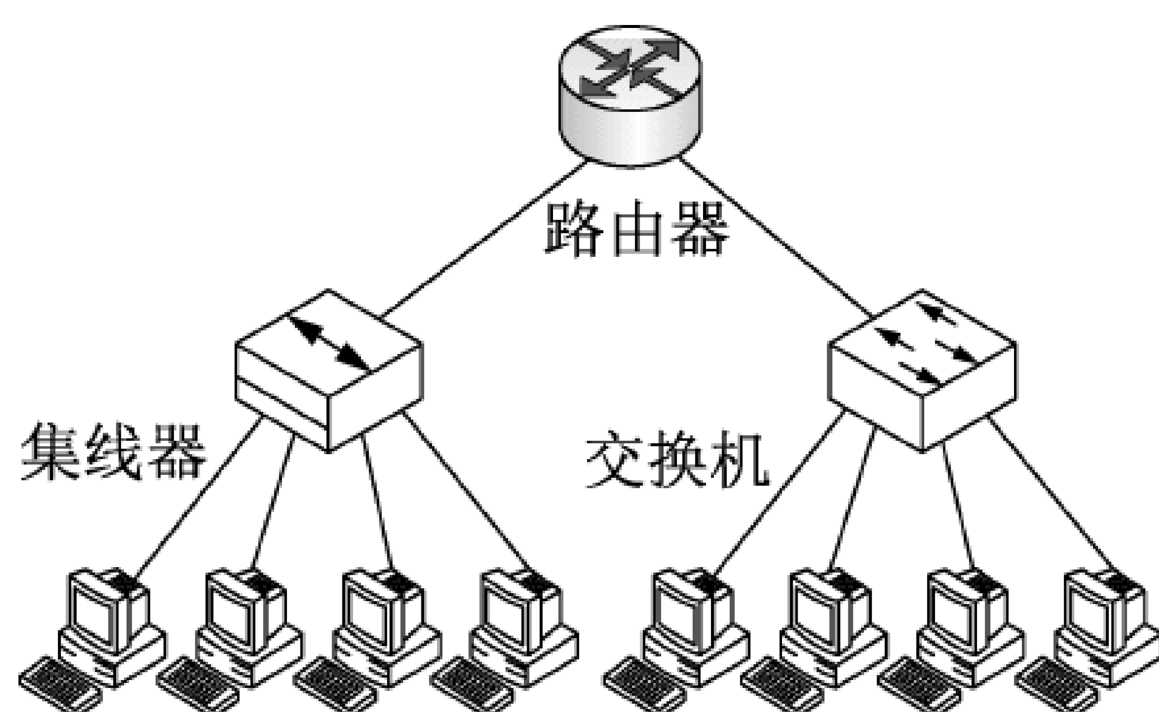
(3) 网络设备。主要指一些硬件设备。

(4) 通信介质。数据的传输媒体。

(5) 网络软件。主要包括底层协议软件、网络操作系统等。

10.3.2 典型例题分析

例 1 网络配置如下图所示，其中使用了一台路由器、一台交换机和一台集线器，对于这种配置，下面的论断中正确的是__(66)__。(2013 年上半年试题 66)



(66) A. 2 个广播域和 2 个冲突域　　B. 1 个广播域和 2 个冲突域
　　C. 2 个广播域和 5 个冲突域　　D. 1 个广播域和 8 个冲突域

解析：路由器工作在网络层，可以识别网络层的地址——IP 地址，有能力过滤第 3 层的广播消息。实际上，除非做特殊配置；否则路由器从不转发广播类型的数据包。因此，路由器的每个端口所连接的网络都独自构成一个广播域。

集线器上构成一个物理上的星型连接。但实际上，在集线器内部，各接口都是通过背板总线连接在一起的，在逻辑上仍构成一个共享的总线。因此，集线器及其所有接口所接的主机共同构成了一个冲突域和一个广播域。

交换机是通过为需要通信的两台主机直接建立专用的通信信道来增加可用带宽的。从这个角度来讲，交换机相当于多端口网桥。交换机上的每个接口都是自己的一个冲突域。交换机及其所有接口所连接的主机共同构成了一个广播域。

答案：C

例 2 集线器与网桥的区别是__(66)__。(2015 年下半年试题 66)

(66) A. 集线器不能检测发送冲突，而网桥可以检测冲突
　　B. 集线器是物理层设备，而网桥是数据链路层设备
　　C. 网桥只有两个端口，而集线器是一种多端口网桥
　　D. 网桥是物理层设备，而集线器是数据链路层设备

解析：集线器是物理层设备，可视为一种特殊的中继器，用于扩大网络；网桥是数据链路层设备，用于连接两个局域网网段。确切地讲，网桥工作在 MAC 子层，只要两个网络的 MAC 子层以上的协议相同，都可以用网桥互联。

答案：B

例 3　下面关于集线器与交换机的描述中，错误的是 (68) 。(2011 年上半年试题 68)

(68) A．交换机是一种多端口网桥

B．交换机的各个端口形成一个广播域

C．集线器的所有端口组成一个冲突域

D．集线器可以起到自动寻址的作用

解析：集线器工作在物理层，其主要功能是对接收到的信号进行再生整形放大，以扩大网络的传输距离，同时把所有节点集中在以它为中心的节点上。集线器采用广播方式发送数据，也就是说，当它要向某节点发送数据时，不是直接把数据发送到目的节点，而是把数据包发送到与集线器相连的所有节点。连接到一个集线器的所有节点共享一个冲突域。但是集线器没有自动寻址的作用。

交换机是一种工作在数据链路层的网络设备，它是基于 MAC 地址识别，完成转发数据帧功能的一种网络连接设备。交换机可以在它的多个端口之间建立多个并发连接。但是，交换机没有过滤广播通信的功能。如果交换机收到一个广播数据包后，它会向其所有的端口转发此广播数据包。因此，交换机及其所有接口所连接的主机共同构成了一个广播域。

答案：D

例 4　下列网络互联设备中，属于物理层的是(66)，属于网络层的是(67)。(2009 年下半年试题 66、67)

(66)～(67)　A．中继器　　B．交换机　　C．路由器　　D．网桥

解析：本题考查网络互联设备知识点。

物理层的互联设备有中继器(Repeater)和集线器(Hub)，数据链路层的互联设备有网桥(Bridge)和交换机(Switch)，网路层的互联设备是路由器。

答案：(66) A　(67) C

例 5　在 Windows 系统中设置默认路由的作用是(67)。(2009 年上半年试题 67)

(67) A．当主机接收到一个访问请求时首先选择的路由

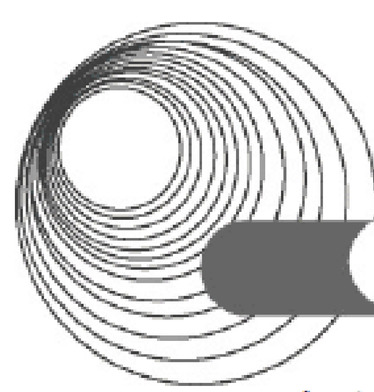
B．当没有其他路由可选时最后选择的路由

C．访问本地主机的路由

D．必须选择的路由

解析：默认路由是一种特殊的静态路由，指的是当路由表中与包的目的地址之间没有匹配的表项时路由器能够作出的选择。如果没有默认路由器，那么目的地址在路由表中没有匹配表项的包将被丢弃。默认路由在某些时候非常有效，当存在末梢网络时，默认路由会大大简化路由器的配置，减轻管理员的工作负担，提高网络性能。

路由器在没有配置路由时，只能实现与它直连的网络间的通信，为了实现在更大范围的网络间通信，需要进行路由配置。路由包括静态路由、默认路由和动态路由几类。静态路由是一种由网管手工配置的路由路径，网管必须了解路由器的拓扑连接，通过手工方式指定路由路径，而且在网络拓扑发生变动时，也需要网管手工修改路由路径。默认路由也是一种由网管手工配置的路由路径，它使路由器把所有地址不能识别的数据包通过指定的



路径发送出去，由其他路由器进行处理。默认路由可看作静态路由的特例，而且维护代价较低。动态路由是一种通过某种路由协议，由路由器自学习到的路由，它不需要手工配置，而且可自动随着网络环境的变化而变化，维护代价很低，特别适合大范围的路由。

答案：B

10.3.3 同步练习

1. ADSL 是一种宽带接入技术，这种技术使用的传输介质是_____。
   A．电话线　B．CATV 电缆　C．基带同轴电缆　D．无线通信网
2. 下面关于网络系统设计原则的论述，正确的是_____。
   A．应尽量采用先进的网络设备获得最高的网络性能
   B．网络总体设计过程中，只需要考虑近期目标即可，不需要考虑扩展性
   C．系统应采用开放的标准和技术
   D．网络需求分析独立于应用系统的需求分析
3. 关于路由器，下列说法中错误的是_____。
   A．路由器可以隔离子网，抑制广播风暴
   B．路由器可以实现网络地址转换
   C．路由器可以提供可靠性不同的多条路由选择
   D．路由器只能实现点对点的传输
4. 与多模光纤相比较，单模光纤具有_____等特点。
   A．较高的传输率、较长的传输距离、较高的成本
   B．较低的传输率、较短的传输距离、较高的成本
   C．较高的传输率、较短的传输距离、较低的成本
   D．较低的传输率、较长的传输距离、较低的成本

10.3.4 同步练习参考答案

1．A　2．C　3．D　4．A

10.4 网络的协议与标准

10.4.1 考点辅导

10.4.1.1 网络的标准

1．电信标准

(1) V 系列：主要针对调制解调器的标准。

(2) X 系列：应用于广域网。

① X.1～X.39：标准应用于终端形式、接口、服务设施和设备。最著名的标准是 X.25，规定了数据包装和传送的协议。

② X.40～X.199：标准管理网络结构、传输、发信号等。

2. 相关国际标准的制定机构

(1) ISO——国际标准化组织。
(2) ANSI——美国国家标准学会。
(3) NIST——美国国家标准和技术研究所。
(4) IEEE——电气和电子工程师协会。
(5) EIA——电子工业协会。

10.4.1.2　局域网协议

1. LAN 模型

ISO/OSI 的 7 层参考模型，其本身不是一个标准，在制定具体网络协议和标准时，要将 OSI/RM 参考模型作为“参照基准”。

(1) 物理层。主要处理在物理链路上发送、传递和接收的非结构化的比特流。

(2) 介质访问控制(MAC)层。控制对传输介质的访问、介质的访问控制和对信道资源的分配，实现帧的寻址和识别，完成帧检测序列和检验等功能。

(3) 逻辑链路控制(LLC)层。提供可靠的信道、数据帧的封装和拆除，为高层提供网络服务的逻辑接口，能够实现差错控制和流量控制。

2. 以太网(IEEE 802.3 标准)

目前以太网主要包括 3 种类型：IEEE 802.3 中定义的标准局域网，速度为 10Mb/s，传输介质为细同轴电缆；IEEE 802.3u 中定义的快速以太网，速度为 100Mb/s，传输介质为双绞线；IEEE 802.3z 中定义的千兆以太网，速度为 1000Mb/s，传输介质为光纤或双绞线。

3. 令牌环网(IEEE 802.5)

令牌环是环状网中最普遍采用的介质访问控制，它是用于环状网结构的分布式介质访问控制，其流行性仅次于以太网。令牌环网的传输介质主要基于屏蔽双绞线、非屏蔽双绞线两种；拓扑结构可以有多种，如环状、星状、总线；编码方法为差分曼彻斯特编码。

工作过程：首先，令牌环网在网络中传递一个很小的帧，称为“令牌”，拥有它的工作站才有权力发送信息；令牌在网络上顺序传递；获得一个空令牌的工作站将要发送的信息附加到后边发往下一站，直到目的站，将令牌释放；如果经过的令牌不为空，则等待令牌释放。

4. FDDI(光纤分布式数据接口)

类似令牌环网的协议，用光纤作为传输介质，数据传输可达 100Mb/s，环路长度可扩展到 200km。连接的站点数可以达到 1000 个。它采用双环体系结构，两环上的信息反方向流动。双环中的一环称为主环，另一环称为次环。

10.4.1.3 广域网协议

1. 点对点协议

点对点协议(PPP)主要通过拨号或专线方式建立点对点连接发送数据，使其成为各种主机、网桥和路由器之间简单连接的一种共通的解决方案。优点在于简单、具备用户验证能力、可以解决 IP 分配等。

2. 数字用户线

xDSL 是各种数字用户线的统称，包括：ADSL——不对称数字用户线；SDSL——单对线数字用户环路；IDSL——ISDN 用的数字用户线；RADSL——速率自适应非对称型数字用户线；VDSL——甚高速数字用户线等。

3. 数字专线

数字数据网(Digital Data Network，DDN)是采用数字传输信道传输数据信号的通信网，可提供点对点、点对多点透明传输的数据专线出租线路，为用户传输数据、图像、声音等信息。数字数据网是以光纤为中继干线网络，组成 DDN 的基本单位是节点，节点间通过光纤连接，构成网状的拓扑结构。它是采用固定连接的方式，直接进入电信的 DDN 网络。

4. 帧中继

帧中继(FR)是在用户网络接口之间提供用户信息流的双向传送，并保持顺序不变的一种承载业务。用户信息以帧为单位进行传输，并对用户信息流进行统计复用。帧中继提供一种简单的面向连接的虚电路分组服务。帧中继的优点是：降低网络互联费用，简化网络功能，提高网络性能，采用国际标准，各厂商产品相互兼容。

5. 异步传输模式

异步传输模式(ATM)是一种面向分组的快速分组交换模式，使用了异步时分复用技术，将信息流分割成固定长度的信元。

ATM 参考模型由 4 层构成。

(1) 用户层：由用户平面、控制平面和管理平面组成。

(2) ATM 适配层：负责将用户层的信息转换成 ATM 网络可用的格式。

(3) ATM 层：负责生成信元。

(4) 物理层：负责对信元进行编码，并将其交给物理介质。

6. X.25 协议

X.25 协议在本地和远程之间提供一个全双工、同步的透明信道，并定义了 3 个相互独立的控制层，即物理层、链路层和分组层，它们分别对应于 ISO/OSI 的物理层、链路层和网络层。X.25 是在公共数据网上，以分组方式进行操作的 DTE 和 DCE 之间的接口。

10.4.1.4 TCP/IP 协议簇

TCP/IP 作为 Internet 的核心协议，被广泛应用于局域网和广域网中，目前已成为事实上的国际标准。TCP/IP 包含许多重要的基本特性，这些特性主要表现在 5 个方面，即逻辑编址、路由选择、域名解析、错误检测与流量控制以及对应用程序的支持等。

1. TCP/IP 分层模型

(1) 应用层。处在分层模型的最高层，用户调用应用程序来访问 TCP/IP 互联网络，以享受网络上提供的各种服务。

(2) 传输层。提供应用程序之间的通信服务。这种通信又叫端到端的通信。传输层既要系统地管理数据信息的流动，还要提供可靠的传输服务，以确保数据准确而有序地到达目的地。

(3) 网际层。又称为 IP 层，主要处理机器之间的通信问题。它接受传输层请求，传送某个具有目的地址信息的分组。该层主要完成：把分组封装到 IP 数据报中，填入数据报的首部，使用路由算法选择；处理接收到的数据报；适时发出 ICMP 的差错和控制报文，并处理收到的 ICMP 报文。

(4) 网络接口层。处在 TCP/IP 的最底层，主要负责管理为物理网络准备数据所需的全部服务程序和功能。该层包含设备驱动程序，也可能是一个复杂的使用自己的数据链路协议的子系统。

2. 网络接口层协议

TCP/IP 不包含具体的物理层和数据链路层，只定义了网络接口层作为物理层与网络层的接口规范。这个物理层可以是广域网，如 X.25 公用数据网；可以是局域网，如 Ethernet、Token-Ring 和 FDDI 等。任何物理网络只要按照这个接口规范开发网络接口驱动程序，都能够与 TCP/IP 集成起来。网络接口层处在 TCP/IP 的最底层，主要负责管理为物理网络准备数据所需的全部服务程序和功能。

3. 网际层协议——IP

IP 所提供的服务通常被认为是无连接的和不可靠的。事实上，在网络性能良好的情况下，IP 传送的数据能够完好无损地到达目的地。

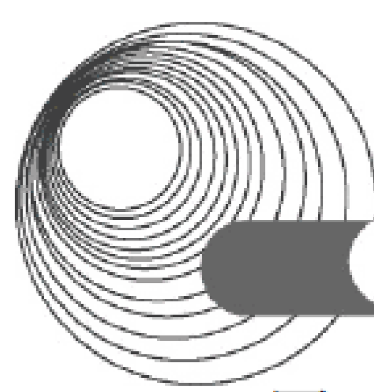
IP 的主要功能包括将上层数据(如 TCP、UDP 数据)或同层的其他数据(如 ICMP 数据)封装到 IP 数据报中；将 IP 数据报传送到最终目的地；为了使数据能够在链路层上进行传输，对数据进行分段；确定数据报到达其他网络中的目的地的路径。

4. ARP 和 RARP

地址解析协议(Address Resolution Protocol，ARP)及反地址解析协议(RARP)是驻留在网际层中的另一个重要协议。ARP 的作用是将 IP 地址转换为物理地址，RARP 的作用是将物理地址转换为 IP 地址。

5. 网际层协议——ICMP

Internet 控制信息协议(Internet Control Message Protocol，ICMP)是网际层的另一个比较重要的协议。由于 IP 是一种尽力传送的通信协议，即传送的数据报可能丢失、重复、延迟或乱序，因此 IP 需要一种避免差错并在发生差错时报告的机制。ICMP 就是一个专门用于发送差错报文的协议。ICMP 定义了 5 种差错报文(源抑制、超时、目的不可达、重定向和要求分段)和 4 种信息报文(回应请求、回应应答、地址屏蔽码请求和地址屏蔽码应答)。IP 在需要发送一个差错报文时要使用 ICMP，而 ICMP 也是利用 IP 来传送报文的。ICMP 是让 IP 更加稳固、有效的一种协议，它使得 IP 传送机制变得更加可靠。而且利用 ICMP 还可以



用于测试因特网，以得到一些有用的网络维护和排错的信息。例如，著名的 ping 工具就是利用 ICMP 报文进行目标是否可达测试。

6. 传输层协议——TCP

TCP(Transmission Control Protocol，传输控制协议)是整个 TCP/IP 协议簇中最重要的协议之一。它在 IP 提供的不可靠数据服务的基础上，为应用程序提供了一个可靠的、面向连接的、全双工的数据传输服务。

TCP 是如何实现可靠性的？最主要和最重要的是 TCP 采用了一个叫重发(Retransmission)的技术。具体来说，在 TCP 传输过程中，发送方启动一个定时器，然后将数据包发出，当接收方收到这个信息就给发送方一个确认(Acknowledgement)信息。而如果发送方在定时器到点之前没收到这个确认信息，就重新发送这个数据包。

7. 传输层协议——UDP

用户数据报协议(User Datagram Protocol，UDP)是一种不可靠的、无连接的协议，可以保证应用程序进程间的通信。与同样处在传输层的面向连接的 TCP 相比较，UDP 是一种无连接的协议，它的错误检测功能要弱得多。可以这样说，TCP 有助于提高可靠性；而 UDP 则有助于提高传输的高速率性。

8. 应用层协议

随着计算机网络的广泛应用，人们也已经有了许多基本的、相同的应用需求。为了让不同平台的计算机能够通过计算机网络获得一些基本的、相同的服务，也就应运而生了一系列应用级的标准，实现这些应用标准的专用协议称为应用级协议，相对于 OSI 参考模型来说，它们处于较高的层次结构，所以也称为高层协议。应用层的协议有 NFS、Telnet、SMTP、DNS、SNMP 和 FTP 等。

10.4.2 典型例题分析

例 1 把网络地址 117.15.32.0/23 划分为 117.15.32.0/27，则得到的子网是 (67) 个。每个子网中可使用的主机地址是 (68) 个。(2013 年上半年试题 67、68)

(67) A. 4　　B. 8　　C. 16　　D. 32

(68) A. 30　　B. 31　　C. 32　　D. 34

解析：题中是将原来掩码为 23 位的地址划分成掩码为 27 位的地址。根据定义，子网的数目为 2^n 个，若将原来掩码为 23 位的地址划分一次变成两个子网，此时，掩码的长度加一位，即 24 位，依次类推，即掩码位数每增加一位，子网的数目变成原来的 2 倍，所以当掩码位数为 27 位时，得到的子网数为 2^{27-23} 个，即 2^4=16 个，所以选 C。因为划分后的子网掩码的位数为 27 位，即子网掩码为 11111111.11111111.11111111.11100000，变成十进制为 255.255.255.224。可以看出，子网划分时网络位向主机位借了 3 位，主机位还剩下 5 位，所以划分出来的子网数为 2 的 5 次方，即 32 个，又因为一个网络地址和一个广播地址不可分配给主机，所以每个子网可用主机地址数为 32−2=30 个。所以选择 A。

答案：(66) C　　(67) A

例 2 (7) 协议在终端设备与远程站点之间建立安全连接。(2015 年上半年试题 7)

(7) A. ARP　　B. Telnet　　C. SSH　　D. WEP

解析：SSH 为 Secure Shell 的缩写，由 IETF 的网络工作小组(Network Working Group)所制定；SSH 为建立在应用层和传输层基础上的安全协议。SSH 是目前较可靠，专为远程登录会话和其他网络服务提供安全性的协议。利用 SSH 协议可以有效防止远程管理过程中的信息泄露问题。

ARP 是地址解析协议；Telnet 是远程登录协议；WEP(Wired Equivalent Privacy)，有线等效保密协议是对在两台设备间无线传输的数据进行加密的方式，用以防止非法用户窃听或侵入无线网络。

答案：C

例 3　DHCP 协议的功能是__(69)__；FTP 使用的传输层协议为__(70)__。(2015 年上半年试题 69、70)

(69) A. WINS 名字解析　　B. 静态地址分配
　　C. DNS 名字登录　　D. 自动分配 IP 地址

(70) A. TCP　　B. IP　　C. UDP　　D. HDLC

解析：DHCP(Dynamic Host Configuration Protocol，动态主机配置协议)是一个局域网的网络协议，使用 UDP 协议工作，主要有两个用途：给内部网络或网络服务供应商自动分配 IP 地址；给用户或者内部网络管理员作为对所有计算机作中央管理的手段。FTP 使用的传输层协议为 TCP，默认情况下 FTP 协议使用 TCP 端口中的 20 和 21 这两个端口，其中 20 用于传输数据，21 用于传输控制信息。

答案：(69) D　(70) A

例 4　建筑物综合布线系统中的园区子系统是指__(69)__。(2012 年上半年试题 69)

(69) A. 由终端到信息插座之间的连线系统　　B. 楼层接线间到工作区的线缆系统
　　C. 各楼层设备之间的互连系统　　D. 连接各个建筑物的通信系统

解析：由终端到信息插座之间的连线系统是工作区子系统；楼层接线间到工作区的线缆系统是水平子系统；各楼层设备之间的互联系统是干线子系统。

答案：D

例 5　TCP 使用的流量控制协议是__(68)__。(2015 年下半年试题 68)

(68) A. 固定大小的滑动窗口协议　　B. 后退 N 帧的 ARQ 协议
　　C. 可变大小的滑动窗口协议　　D. 停等协议

解析：TCP 使用一种窗口(Window)机制来控制数据流。TCP 的窗口以字节为单位进行调整，以适应接收方的处理能力。处理过程如下。

(1) TCP 连接阶段，双方协商窗口尺寸，同时接收方预留数据缓存区。

(2) 发送方根据协商的结果，发送符合窗口尺寸的数据字节流，并等待对方的确认。

(3) 发送方根据确认信息，改变窗口的尺寸，增加或者减少发送未得到确认的字节流中的字节数。调整过程包括：如果出现发送拥塞，发送窗口缩小为原来的一半，同时将超时重传的时间间隔扩大一倍。

答案：C

例 6　ARP 协议属于__(66)__协议，它的作用是__(67)__。(2011 年上半年试题 67)

(66) A. 物理层　　B. 数据链路层　　C. 网络层　　D. 传输层

(67) A．实现 MAC 地址与主机名之间的映射
B．实现 IP 地址与 MAC 地址之间的变换
C．实现 IP 地址与端口号之间的映射
D．实现应用进程与物理地址之间的变换

解析：ARP(Address Resolution Protocol，地址解析协议)属于网络层协议。在实际通信中，物理网络依然利用物理地址进行数据报的传输，IP 地址在物理网络中是不能被识别的。将 IP 地址映射到物理地址的方法有多种，以太网经常使用的便是 ARP 协议，它充分利用了以太网的广播能力，将 IP 地址与物理地址动态绑定。

答案：(66) C　(67) B

例 7　以下 4 种路由中，(69) 路由的子网掩码是 255.255.255.255。(2015 年下半年试题 69)

(69) A．远程网络　B．静态　C．默认　D．主机

解析：主机路由的目的地址是一个完整的主机地址，子网掩码为 255. 255. 255. 255。网络路由目的地址是一个网络地址，主机号部分为 0。

静态路由需要管理员根据实际需要一条条自己手动配置。静态路由中包括目标节点或目标网络的 IP 地址，对应的子网掩码一般为 255. 255. 255. 0；默认路由是一种特殊的静态路由，指的是当路由表中与包的目的地址之间没有匹配的表项时路由器能够做出的选择。默认路由和静态路由的命令格式一样，只是把目的地 IP 和子网掩码改成 0.0.0.0 和 0.0.0.0。

答案：(69) D

例 8　ICMP 协议属于因特网中的 (67) 协议，ICMP 协议数据单元封装在 (68) 中传送。(2014 年下半年试题 67、68)

(67) A．数据链路层　B．网络层　C．传输层　D．会话层
(68) A．以太帧　B．TCP 段　C．UDP 数据报　D．IP 数据报

解析：ICMP 是 TCP/IP 协议族的一个子协议，属于网络层协议，主要用于在主机与路由器之间传递控制信息，包括报告错误、交换受限控制和状态信息等。和 TCP、UDP、IGMP 等协议一样，ICMP 协议使用 IP 数据报传送数据。

答案：(67) B　(68) D

例 9　帧中继作为一种远程接入方式有许多优点，下面的选项中错误的是 (69) 。(2010 年下半年试题 69)

(69) A．帧中继比 X.25 的通信开销少，传输速度更快
B．帧中继与 DDN 相比，能以更灵活的方式支持突发式通信
C．帧中继比异步传输模式能提供更高的数据速率
D．租用帧中继虚电路比租用 DDN 专线的费用低

解析：帧中继的帧长可变，数据传输速率在 2～45Mb/s 之间。异步传输模式把用户数据组织成 53B 长的信元，由于信元长度固定，可以进行高速的处理和交换，典型的数据速率为 150Mb/s。可见，异步传输模式能提供更高的数据速率，选项 C 是错误的。

答案：C

例 10　以下关于 URL 的叙述中，不正确的是 (68) 。(2015 年上半年试题 68)

(68) A．使用 www. abc. com 和 abc. com 打开的是同一页面

B．在地址栏中输入 www.abc.com 默认使用 http 协议

C．www.abc.com 中的“www”是主机名

D．www.abc.com 中的“abc.com”是域名

解析：www. abc. com 中的“abc. com”是域名，在该域名下可以创建指向不同主机的域名，如 www 主机名，www 的域名其实只是不带 www 的主域名下的一个特殊的二级域名。

答案：A

例 11　POP3 协议采用 (67) 模式，客户端代理与 POP3 服务器通过建立 TCP 连接来传送数据。(2015 年下半年试题 67)

(67) A．Browser/Server　　B．Client/Server

C．Peer to Peer　　D．Peer to Server

解析：POP 适用于 C/S 结构的脱机模型的电子邮件协议，目前已发展到第三版，称为 POP3。POP 协议支持“离线”邮件处理，其具体过程是：邮件发送到服务器上，电子邮件客户端调用邮件客户机程序以连接服务器，并下载所有未阅读的电子邮件。

答案：B

例 12　PPP 中的安全认证协议是 (66) ，它使用三次握手的会话过程传送密文。(2014 年下半年试题 66)

(66) A. MDS　　B. PAP　　C. CHAP　　D. NCP

解析：(1)　安全认证介绍。

① PPP 的 NCP 可以承载多种协议的三层数据包。

② PPP 使用 LCP 控制多种链路的参数(建立、认证、压缩、回拨)。

(2) PPP 的认证类型。

① PPP 的 PAP 认证是通过二次握手建立认证(明文不加密)。

② PPP 的 CHAP 挑战握手认证协议，通过三次握手建立认证(密文采用 MDS 加密)。

③ PPP 的双向验证，采用的是 CHAP 的主验证风格。

④ PPP 的加固验证，采用的是两种(PAP、CHAP)验证同时使用。

答案：C

10.4.3　同步练习

1.　TCP/IP 协议在多个层引入了安全机制，其中 TLS 协议位于______。

A．数据链路层　　B．网络层　　C．传输层　　D．应用层

2.　运行 Web 浏览器的计算机与网页所在的计算机要建立(1)连接，采用(2)协议传输网页文件。

(1)　A．UDP　　B．TCP　　C．IP　　D．RIP

(2)　A．HTTP　　B．HTML　　C．ASP　　D．RPC

3.　关于 ARP 表，以下描述中正确的是______。

A．提供常用目标地址的快捷方式来减少网络流量

B．用于建立 IP 地址到 MAC 地址的映射

C．用于在各个子网之间进行路由选择

D．用于进行应用层信息的转换

4. ARP 的作用是 (1) ，ARP 报文封装在 (2) 中传送。

(1) A．由 IP 地址查找对应的 MAC 地址

B．由 MAC 地址查找对应的 IP 地址

C．由 IP 地址查找对应的端口号

D．由 MAC 地址查找对应的端口号

(2) A．以太帧　　B．IP 数据报　　C．UDP 报文　　D．TCP 报文

5. 802.11 标准定义的分布式协调功能采用了______协议。

A．CSMA/CD　　B．CSMA/CA　　C．CDMA/CD　　D．CDMA/CA

6. 以下不属于中间件技术的是______。

A．JavaRMI　　B．CORBA　　C．DCOM　　D．JavaApplet

7. DHCP 客户端可以从 DHCP 服务器获得______。

A．DHCP 服务器的地址和 Web 服务器的地址

B．DNS 服务器的地址和 DHCP 服务器的地址

C．客户端地址和邮件服务器地址

D．默认网关的地址和邮件服务器地址

10.4.4 同步练习参考答案

1．C　　2．(1) B　(2) A

3．B　　4．(1) A　(2) A

5．B　　6．D

7．B

10.5 Internet 及应用

10.5.1 考点辅导

10.5.1.1 Internet 概述

从用户的角度来看，整个 Internet 在逻辑上是统一的、独立的，在物理上则由不同的网络互联而成。从技术角度看，Internet 本身不是某一种具体的物理网络技术，它是能够互相传递信息的众多网络的一个统称，或者说它是一个网间网，只要人们进入了这个互联网，就是在使用 Internet。

在 Internet 中，分布着一些覆盖范围很广的大网络，这种网络称为“Internet 主干网”，它们一般属于国家级的广域网。例如，我国的 CHINANET 和 CERNET 等就是中国的 Internet 主干网。主干网一般只延伸到一些大城市或重要地区，在那里设立主干网节点。每一个主干网节点可以通过路由器将广域网与局域网连接起来，一个节点还可以通过另外的路由器

与其他局域网再互联，由此形成一种网状结构。

10.5.1.2　Internet 地址

1. 域名

一个完整、通用的层次型主机域名由 4 部分组成，即计算机主机名.本机名.组名.最高层域名。

2. IP 地址

(1) A 类：网络地址占 1B，最高位为 0；主机地址占 3B。子网掩码为 255.0.0.0。
(2) B 类：网络地址占 2B，最高位为 10；主机地址占 2B。子网掩码为 255.255.0.0。
(3) C 类：网络地址占 3B，最高位为 110；主机地址占 1B。子网掩码为 255.255.255.0。
(4) D 类：用于组播。最高位为 1110。
(5) E 类：实验保留。最高位为 1111。

3. NAT 技术

因特网面临 IP 地址短缺的问题。解决这个问题有所谓长期的和短期的两种解决方案。长期的解决方案就是使用具有更大地址空间的 IPv6 协议，网络地址翻译(Network Address Translators，NAT)是短期的解决方案中的一种。NAT 的实现主要有两种形式，即动态地址翻译(Dynamic Address Translation，DAT)和 m∶1 翻译(这种技术也叫做伪装)。

4. IPv6 简介

1) IPv6 数据包的格式

IPv6 数据包有一个 40B 的基本首部(Base Header)，其后可允许有 0 个或多个扩展首部(Extension Header)，再后面是数据。每个 IPv6 数据包都是从基本首部开始的。

2) IPv6 的地址表示

一般来讲，一个 IPv6 数据包的目的地址可以是以下 3 种基本类型地址之一。

(1) 单播(Unicast)：传统的点对点通信。

(2) 多播(Multicast)：一点对多点的通信，数据包交付到一组计算机中的每一个。IPv6 没有采用广播的术语，而是将广播看作多播的一个特例。

(3) 任播(Anycast)：这是 IPv6 增加的一种类型。任播的目的站是一组计算机，但数据包在交付时只交付给其中的一个，通常是距离最近的一个。

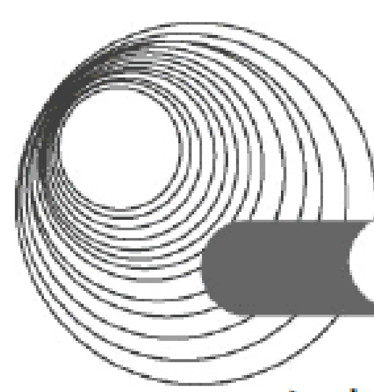
10.5.1.3　Internet 服务

1. DNS 域名服务

DNS 是一种分布式地址信息数据库系统，服务器中包含整个数据库的某部分信息，并供客户查询。域名系统采用的是客户机/服务器模式，整个系统由解析器和域名服务器组成。解析器是客户方，它负责查询域名服务器、解释从服务器返回来的应答、将信息返回给请求方等工作。域名服务器是服务器方，它通常保存着一部分域名空间的全部信息。

2. 远程登录服务

远程登录服务是在 Telnet 协议的支持下，将用户计算机与远程主机连接起来，在远程



主机上运行程序，将相应的屏幕显示传送到本地机器，并将本地的输入送给远程计算机。

3. 电子邮件服务

电子邮件就是利用计算机进行信息交换的电子媒体信件。电子邮件地址的一般格式为“用户名@主机名”。E-mail 系统基于客户机/服务器模式，整个系统由 E-mail 客户软件、E-mail 服务器和通信协议三部分组成。所用协议有简单邮件传送协议(SMTP)和用于接收邮件的 POP3 协议。

4. WWW 服务

WWW 服务是一种交互式图形界面的 Internet 服务，具有强大的信息连接功能。WWW 浏览程序为用户提供基于超文本传输协议(HTTP)的用户界面，WWW 服务器的数据文件由超文本标记语言(HTML)描述，(HTTP)利用统一资源定位地址(URL)指向超媒体链接，并在文本内指向其他网络资源。一个 URL 包括协议、主机域名、端口号(任选)、目录路径(任选)和一个文件名(任选)几部分。其格式为 scheme://host.Domain[:port]Upath/filename。

5. 文件传输服务

文件传输服务用来在计算机之间传输文件。FTP 是基于客户机/服务器模式的服务系统，它由客户软件、服务器软件和 FTP 通信协议三部分组成。FTP 在客户与服务器的内部建立两条 TCP 连接：一条是控制连接，主要用于传输命令和参数；另一条是数据连接，主要用于传送文件。

10.5.2 典型例题分析

例 1 在 ASP 的内置对象中，(67) 对象可以修改 cookie 中的值。(2012 年下半年试题 67)

(67) A．request　　B．response　　C．application　　D．session

解析：request 对象访问任何用 HTTP 请求传递的信息，包括从 HTML 表格用 POST 方法或 GET 方法传递的参数、cookie 和用户认证。Request 对象使您能够访问发送给服务器的二进制数据，如上载的文件。response 对象控制发送给用户的信息，包括直接发送信息给浏览器、重定向浏览器到另一个 URL 或设置 cookie 的值。application 对象使给定应用程序的所有用户共享信息。session 对象存储特定的用户会话所需的信息。

答案：B

例 2 IP 地址块 155.32.80.192/26 包含了 (66) 个主机地址，以下 IP 地址中，不属于这个网络的地址是 (67) 。(2014 年上半年试题 66、67)

(66) A. 15　　B. 32　　C. 62　　D. 64

(67) A. 155.32.80.202　　B.155.32.80.195

C.155.32.80.253　　D.155.32.80.191

解析：155.32.80.192/26 表示 32 位长度的 IP 地址中，前 26 位是网络前缀，后 6 位是主机号，因此包含的主机地址个数为 $2^6-2=62$，主机地址范围为 155.32.80.193~155.32.80.254，显然 155.32.80.191 不属于这个网络。

答案：(66) C　(67) D

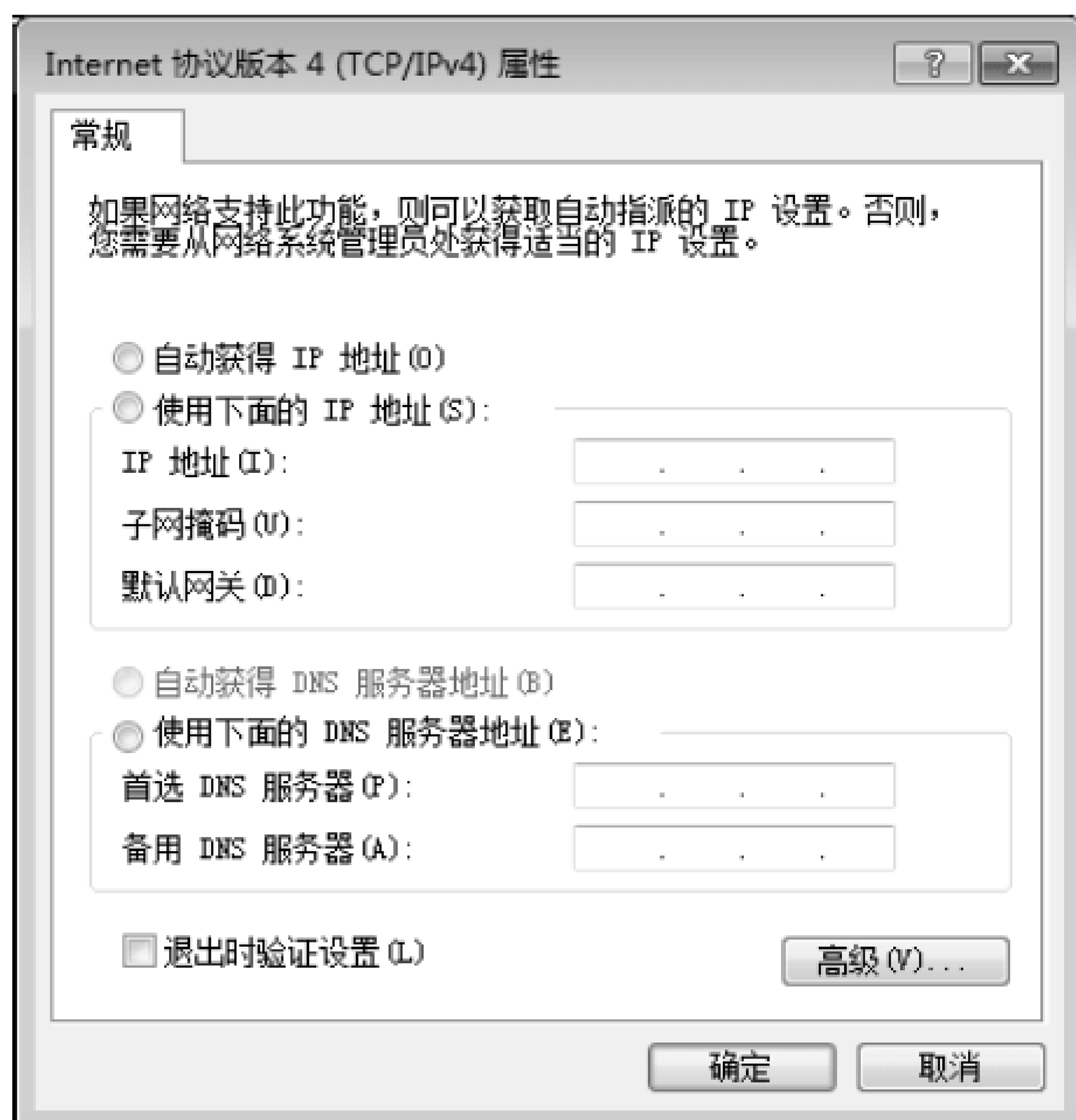
例 3　某用户在使用校园网中的一台计算机访问某网站时，发现使用域名不能访问该网站，但是使用该网站的 IP 地址可以访问该网站，造成该故障产生的原因有很多，其中不包括<u>(69)</u>。(2014 年上半年试题 69)

(69) A．该计算机设置的本地 DNS 服务器工作不正常
　　 B．该计算机的 DNS 服务器设置错误
　　 C．该计算机与 DNS 服务器不在同一子网
　　 D．本地 DNS 服务器网络连接中断

解析：使用域名无法访问网站，而使用 IP 地址可以访问网站，说明域名无法解析成 IP 地址。大部分情况下，企业网的 DNS 服务器放置在防火墙的 DMZ 网段中，或者是企业网直接利用 Internet 中相关 DNS 服务器进行域名解析，它与作为客户端的计算机通常不在同一子网中。这一因素不会造成某台计算机不能使用域名访问外部服务器的现象。

答案：(69) C

例 4　校园网连接运营商的 IP 地址为 202.117.113.3/30，本地网关的地址为 192.168.1.254/24，如果本地计算机采用动态地址分配，在下图中应该如何配置？<u>(68)</u>。(2014 年上半年试题 68)



(68) A．选取“自动获得 IP 地址”
　　 B．配置本地计算机 IP 地址为 192.168.1.×
　　 C．配置本地计算机 IP 地址为 202.115.113.×
　　 D．在网络 169.254.×.×中选取一个不冲突的 IP 地址

解析：由于本地计算机采用动态地址分配，因此选用“自动获得 IP 地址”，DHCP 服务器会为本地计算机自动配置。

答案：A

例 5　A 类网络是很大的网络，每个 A 类网络中可以有<u>(66)</u>个网络地址。实际使用中必须把 A 类网络划分为子网，如果指定的子网掩码为 255.255.192.0，则该网络被划分为<u>(67)</u>个子网。(2011 年下半年试题 66、67)

(66) A．210　　B．212　　C．220　　D．224
(67) A．128　　B．256　　C．1024　　D．2048

解析：A 类网络用第一个字节表示网络地址，最高位为 0，余下的 7 位为真正的网络地址，而 127.0.0.0 网络地址有特殊的用途，A 类网络地址可以支持 126 个网络。A 类网络地址的后 24 位表示主机号，所以每个 A 类网络中可以有 2^{24} 个主机地址。

子网掩码中全 1 部分对应于网络号，255.255.192.0 的二进制表示为 11111111.11111111.11000000.00000000，子网部分借用了 A 类网络主机号的前 10 位，所以可以确定该网络被划分为 2^{10}=1024 个子网。

答案：(66) D　(67) C

例 6　根据下图所示的输出信息，可以确定的是 (9) 。(2015 年下半年试题 9)

```
    C:\> netstat –n
    Active Connections
Proto  Local Address          Foreign Address       State
TCP    192.168.0.200:2011     202.100.112.12:443    ESTABLISHED
TCP    192.168.0.200:2038     100.29.200.110:110    TIME_WAIT
TCP    192.168.0.200:2052     128.105.129.30:80     ESTABLISHED
```

(9) A．本地主机正在使用的端口号是公共端口号
　B．192.168.0.200 正在与 128.105.129. 30 建立连接
　C．本地主机与 202.100.112.12 建立了安全连接
　D．本地主机正在与 100.29.200.110 建立连接

解析：netstat – n 命令用于显示所有已建立的有效连接。连接状态如下。

LISTEN：侦听来自远方的 TCP 端口的连接请求。

SYN-SENT：在发送连接请求后等待匹配的连接请求。

SYN-RECEIVED：在收到和发送一个连接请求后等待对方对连接请求的确认。

ESTABLISHED：代表一个打开的连接。

FIN-WAIT-1：等待远程 TCP 连接中断请求，或先前的连接中断请求的确认。

FIN-WAIT-2：从远程 TCP 等待连接中断请求。

CLOSE-WAIT：等待从本地用户发来的连接中断请求。

CLOSING：等待远程 TCP 对连接中断的确认。

LAST-ACK：等待原来的发向远程 TCP 的连接中断请求的确认。

TIME-WAIT：等待足够的时间以确保远程 TCP 接收到连接中断请求的确认。

CLOSED：没有任何连接状态。

本机使用 3 个不同的端口号：通过 2011 端口与 202.100.112.12 建立了安全连接；通过 2052 端口与 128.105.129.30 建立了安全连接；通过 2038 端口正在与 100.29.200.110 进行连接中断。

答案：C

例 7　在 IE 浏览器中，安全级别最高的区域设置是 (9) 。(2011 年上半年试题 9)

(9) A．Internet　　B．本地 Intranet　　C．可信站点　　D．受限站点

解析：在 IE 浏览器的 Internet 选项卡中可以设置的区域有 Internet、本地 Intranet、可信站点和受限站点。Internet 区域适用于 Internet 网站，但不适用于列在信任和受限制区域中

的网站，该区域的安全级别默认为中—高。本地 Intranet 区域适用于本地网络(Intranet)上的网站，安全级别默认为中—低。可信站点区域包含自己信任的对计算机或文件没有损害的网站，安全级别默认为中。受限站点区域适用于可能会损害计算机或文件的网站，安全级别默认为高。可见，安全级别最高的区域设置是受限站点。

答案：D

例 8　“三网合一”的三网是指 (69) 。(2011 年上半年试题 69)

(69) A．电信网、广播电视网、互联网

B．物联网、广播电视网、电信网

C．物联网、广播电视网、互联网

D．物联网、电信网、互联网

解析：目前，电信网、广播电视网、互联网是主要的用户接入网，这 3 种网络由不同的部门管理。数字技术可以将各种信息都变成数字信号来获取、处理、存储和传输。数字化技术使得这 3 种网络的服务业务相互交叉，3 种网络之间的界限越来越模糊，最终会导致电信网、广播电视网、互联网“三网融合”局面的出现。物联网是最近几年才提出的概念，目前主要处于研究阶段，应用不是很多。

答案：A

例 9　Outlook Express 作为邮件代理软件有诸多优点，以下说法中错误的是 (7) 。(2010 年上半年试题 7)

(7) A．可以脱机处理邮件

B．可以管理多个邮件账号

C．可以使用通讯簿存储和检索电子邮件地址

D．不能发送和接收安全邮件

解析：Outlook Express 能够发送和接收安全邮件，可使用数字标识对邮件进行数字签名和加密。对邮件进行数字签名可以使收件人确认邮件确实是发送的，而加密邮件则保证只有期望的收件人才能阅读该邮件。

答案：D

例 10　标记语言用一系列约定好的标记来对电子文档进行标记，以实现对电子文档的语义、结构及格式的定义。 (49) 不是标记语言。(2010 年上半年试题 49)

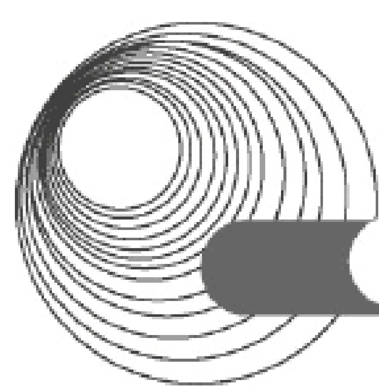
(49) A．HTML　　B．XML　　C．WML　　D．PHP

解析：HTML(HyperText Marked Language，超文本标记语言)，用于互联网的信息表示。用 HTML 编写的超文本文档成为 HTML 文档，它能够独立于各种操作系统平台。

XML(Extensible Markup Language，可扩展的标记语言)丰富了 HTML 的描述功能，可以描述非常复杂的 Web 页面，如复杂的数字表达式、化学方程式等。XML 的特点是结构化、自描述、可扩展和浏览器自适应等。

用于 WAP 的标记语言就是 WML(Wireless Markup Language)，其语法跟 XML 一样，是 XML 的子集。

PHP(Hypertext Preprocessor)是一种在服务器端执行的、嵌入 HTML 文档的脚本语言，其语言风格类似于 C 语言，被网站编程人员广泛运用。



答案：D

例 11 IP 地址块 222.125.80.128/26 包含了 (66) 个可用主机地址，其中最小地址是 (67) ，最大地址是 (68) 。(2010 年上半年试题 66～68)

(66) A．14　　B．30　　C．62　　D．126

(67) A．222.125.80.128　　B．222.125.80.129

C．222.125.80.159　　D．222.125.80.160

(68) A．222.125.80.128　　B．222.125.80.190

C．222.125.80.192　　D．222.125.80.254

解析：/26 表示 IP 地址中前 26 位是网络前缀，后 6 位是主机号，那么可分配的主机地址数是 $2^6-2=64-2=62$ 个，可分配地址范围是 222.125.80.129～222.125.80.190。

答案：(66) C　(67) B　(68) B

例 12 以下 HTML 代码中，创建指向邮箱地址的链接正确的是 (69) 。(2010 年上半年试题 69)

(69) A．test@test.com

B．test@test.com

C．test@test.com

D．test@test.com

解析：创建指向邮箱地址的链接格式为热点文本。

答案：D

例 13 一个 B 类网络的子网掩码为 255.255.224.0，则这个网络被划分成了(66)个子网。(2009 年上半年试题 66)

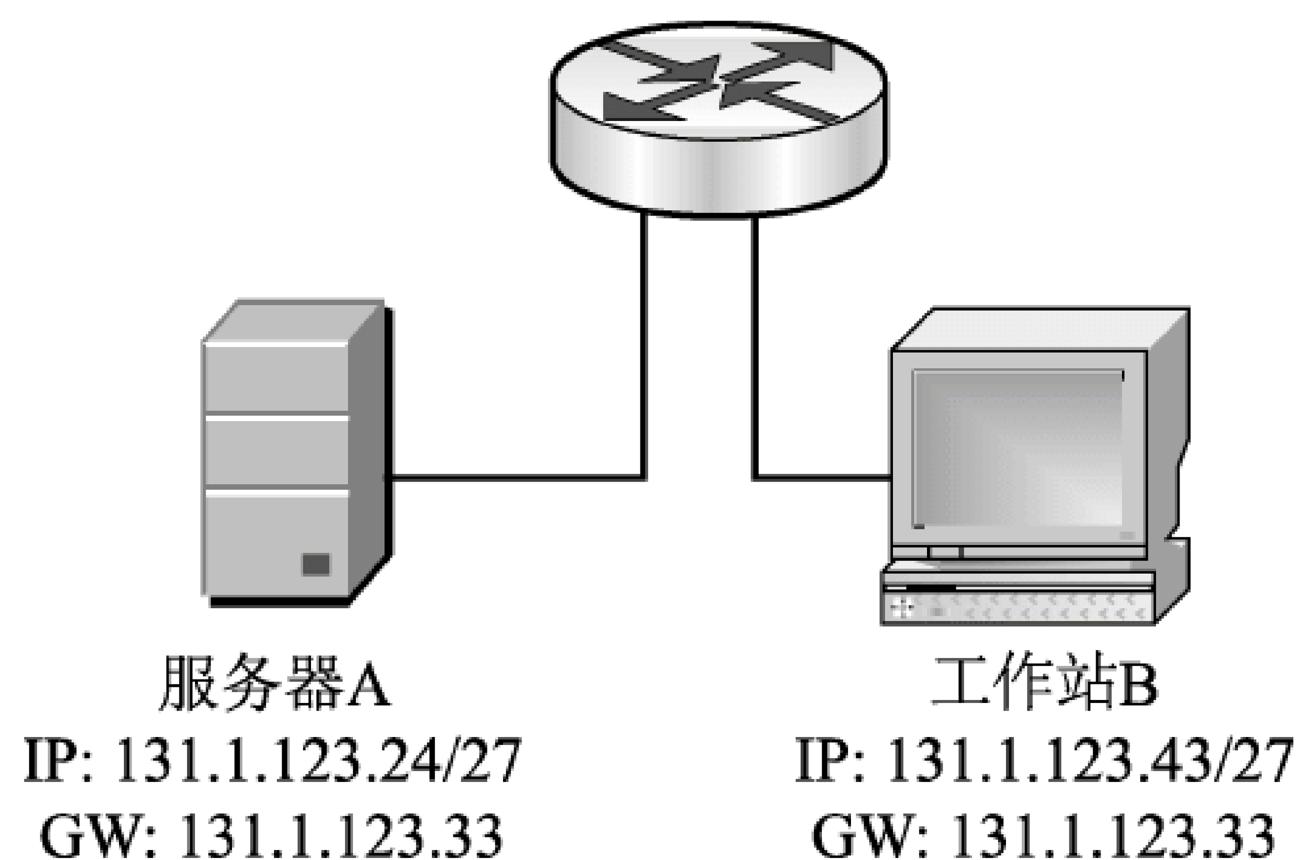
(66) A．2　　B．4　　C．6　　D．8

解析：255.255.224.0 转换成二进制为 1111 1111.1111 1111.1110 0000.0000 0000。

其中 B 类网络的网络号为 255.255.0.0，主机号中的 3 位用来划分子网，即 $2^3=8$，标准答案选 D。但是可用的子网要再减去全为 0 和全为 1 的子网号，即有 6 个子网可用。

答案：D

例 14 在下图所示的网络配置中，发现工作站 B 无法与服务器 A 通信。 (66) 故障影响了两者互通。(2015 年上半年试题 66)



(66) A．服务器 A 的 IP 地址是广播地址

B．工作站 B 的 IP 地址是网络地址

C．工作站 B 与网关不属于同一子网

D．服务器 A 与网关不属于同一子网

解析：服务器 A 的 IP 地址最后一个字段为 24，对应的二进制数为 0001 1000，工作站 B 的 IP 地址最后一个字段为 43，对应的二进制数为 0010 1011，网关的 IP 地址最后一个字段为 33，对应的二进制数为 0010 0001，其中，前 3 位对应子网号，后 5 位对应网络号。可见服务器 A 与网关不属于同一子网。

答案：D

例 15　随着网站知名度不断提高，网站访问量逐渐上升，网站负荷越来越重，针对此问题，一方面可通过升级网站服务器的软硬件，另一方面可以通过集群技术，如 DNS 负载均衡来解决。在 Windows 的 DNS 服务器中通过 (70) 操作可以确保域名解析并实现负载均衡。

(70) A. 启用循环，启动转发器指向每个 Web 服务器

B. 禁止循环，启动转发器指向每个 Web 服务器

C. 禁止循环，添加每个 Web 服务器的主机记录

D. 启用循环，添加每个 Web 服务器的主机记录

解析：通过 DNS 服务器实现网络负载均衡则是一种保证用户网络访问的方式。其方法有启用循环，添加每个 Web 服务器的主机记录。

答案：D

10.5.3　同步练习

1.　下面的选项中，属于本地回路地址的是_____。

A．120.168.10.1　　B．10.128.10.1　　C．127.0.0.1　　D．172.16.0.1

2.　Internet 上的 DNS 服务器中保存有_____。

A．主机名　　B．域名到 IP 地址的映射表

C．所有主机的 MAC 地址　　D．路由表

3.　某银行为用户提供网上服务，允许该用户通过浏览器管理自己的银行账户信息。为保障通信的安全性，Web 服务器可选的协议是_____。

A．POP　　B．SNMP　　C．HTTP　　D．HTTPS

4.　______不属于电子邮件协议。

A．POP3　　B．SMTP　　C．IMAP　　D．MPLS

5.　某客户端在采用 ping 命令检测网络连接故障时，发现可以 ping 通 127.0.0.1 及本机的 IP 地址，但无法 ping 通同一网段内其他工作正常的计算机的 IP 地址，说明该客户端故障是______。

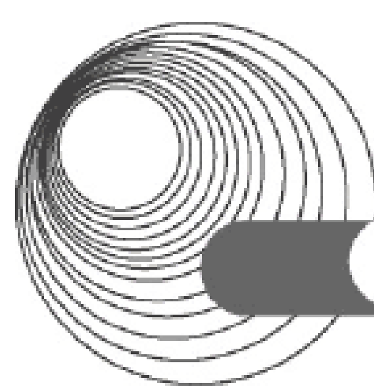
A．TCP/IP 不能正常工作　　B．本机网卡不能正常工作

C．本机网络接口故障　　D．本机 DNS 服务器地址设置错误

6.　用户可以通过 http://www.a.com 和 http://www.b.com 访问在同一台服务器上____不同的两个 Web 站点。

A．IP 地址　　B．端口号　　C．协议　　D．虚拟目录

7.　在 FTP 中，控制连接是由_____主动建立的。



A．服务器端　　B．客户端　　C．操作系统　　D．服务提供商

10.5.4 同步练习参考答案

1．C　　2．B　　3．D　　4．D
5．C　　6．A　　7．B

10.6 网络安全

10.6.1 考点辅导

10.6.1.1 网络安全概述

计算机网络安全是指计算机、网络系统的硬件、软件以及系统中的数据受到保护，不因偶然的或恶意的原因而遭到破坏、更改、泄露，确保系统能连续和可靠地运行，使网络服务不中断。广义地说，凡是涉及网络上信息的保密性、完整性、可用性、真实性和可控性的相关技术和理论，都是网络安全所要研究的领域。

(1) 网络安全涉及的主要内容包括运行系统安全、信息系统安全、信息传播安全和信息内容安全。

(2) 信息系统对安全的基本需求有保密性、完整性、可用性、可控性和可核查性。

(3) 网络安全威胁类别有物理威胁、网络攻击、身份鉴别、编码威胁和系统漏洞。

10.6.1.2 网络的信息安全

1．信息的存储安全

信息的存储安全包括以下内容。

(1) 用户的标识与验证：限制访问系统的人员。

(2) 用户存取权限限制：限制进入系统的用户所能做的操作。

(3) 系统安全监控：建立一套安全监控系统，全面监控系统的活动。

(4) 病毒防治：网络服务器必须加装网络病毒自动检测系统。

由于计算机病毒具有隐蔽性、传染性、潜伏性、触发性和破坏性等特点，所以需要建立计算机病毒防治管理制度。

① 经常从软件供应商网站下载、安装安全补丁程序和升级杀毒软件。

② 定期检查敏感文件。对系统的一些敏感文件定期进行检查，保证及时发现已感染的病毒和黑客程序。

③ 使用高强度的口令。尽量选择难以猜测的口令，对不同的账号选用不同的口令。

④ 经常备份重要数据，要做到每天坚持备份。

⑤ 选择、安装经过公安部认证的防病毒软件，定期对整个硬盘进行病毒检测、清除工作。

⑥ 可以在计算机和因特网之间安装使用防火墙，提高系统的安全性。

⑦ 当计算机不使用时，不要接入因特网，一定要断掉连接。

⑧ 重要的计算机系统和网络一定要严格与因特网物理隔离。

⑨ 不要打开陌生人发来的电子邮件，无论它们有多么诱人的标题或者附件。同时也要小心处理来自熟人的邮件附件。

⑩ 正确配置系统和使用病毒防治产品。正确配置系统，充分利用系统提供的安全机制，提高系统防范病毒的能力，减少病毒侵害事件。了解所选用防病毒产品的技术特点，正确配置以保护自身系统的安全。

(5) 数据的加密：防止非法窃取或调用。

(6) 计算机网络安全：通过采用安全防火墙系统、安全代理服务器、安全加密网关等实现网络信息安全的最外一层防线。

2. 信息的传输安全

信息的传输加密是面向线路的加密措施，有以下 3 种。

(1) 链路加密。只对两个节点之间的通信信道线路上所传输的信息进行加密保护。

(2) 节点加密。加、解密都在节点中进行，即每个节点里装有加、解密的保护装置，用于完成一个密钥向另一个密钥的转换。

(3) 端—端加密。为系统网络提供从信息源到目的地传送的数据的加密保护，可以是从主机到主机、终端到终端、终端到主机或到处理进程，或从数据的处理进程到处理进程，而不管数据在传送过程中经过了多少中间节点，数据均不会被解密。

10.6.1.3　防火墙技术

防火墙(Firewall)是建立在内外网络边界上的过滤封锁机制，它认为内部网络是安全和可信赖的，而外部网络被认为是不安全和不可信赖的。防火墙的作用是防止不希望的、未经授权的数据包进出被保护的内部网络，通过边界控制强化内部网络的安全策略。

1. 防火墙的分类

通常可对防火墙进行以下分类。

(1) 包过滤型防火墙。工作在网络层，对数据包的源 IP 及目的 IP 具有识别和控制作用，对于传输层，它只能识别数据包是 TCP 还是 UDP 及所用的端口信息。优点是：对每条传入和传出网络的包实行低水平控制；每个 IP 包的字段都被检查；可以识别和丢弃带欺骗性源 IP 地址的包；是两个网络之间访问的唯一通道；通常被包含在路由器数据包中，不必用额外的系统来处理这个特征。缺点是：不能防范黑客攻击；不支持应用层协议；访问控制粒度太粗糙。

(2) 应用代理网关防火墙。彻底隔断内网与外网的直接通信，内网用户对外网的访问变成防火墙对外网的访问，然后再由防火墙转发给内网用户。优点是：可以检查应用层、传输层和网络层的协议特征，对数据包的检测能力比较强。缺点是：难以配置，处理速度慢。

(3) 状态检测技术防火墙。结合了代理防火墙的安全性和包过滤防火墙的高速度等优点，在不损失安全性的基础上将代理防火墙的性能提高了 10 倍。

2. 典型防火墙的体系结构

一个防火墙系统通常是由过滤路由器和代理服务器组成的。典型防火墙的体系结构包

括包过滤路由器、双宿主主机、被屏蔽主机网关、被屏蔽子网等。

(1) 包过滤路由器。又称屏蔽路由器，它是最简单也是最常用的防火墙。它一般作用在网络层，对进出内部网络的所有信息进行分析，并按照一定的安全策略对进出内部网络的信息进行限制。包过滤的核心就是安全策略即包过滤算法的设计。优点在于速度快，实现方便；缺点是安全性差，兼容性差，没有或只有较少的日志记录能力。

(2) 双宿主主机。它是围绕着至少具有两个网络接口的双宿主主机构成的，每一个接口都连接在物理和逻辑上分离的不同的网段，代理服务器软件在双宿主主机上运行。优点是：堡垒主机运行的系统软件可用于维护系统日志、硬件复制日志、远程日志等，有利于网络管理员的日后检查。缺点是：由于内部网和外部网之间只有一道屏障，双宿主主机首先禁止网络层的路由功能，两个网络之间的通信通过应用代理层来完成，如果一旦黑客侵入堡垒主机并使其具有路由功能，防火墙会变得无用，需要具有强大的身份认证系统，尽量减少防火墙上用户的账户数目，以免堡垒主机被攻破。

(3) 被屏蔽主机网关。由过滤路由器和应用网关组成。过滤路由器的作用是进行包过滤；应用网关的作用是代理服务，即在内部网络和外部网络之间建立两道安全屏障。优点是安全等级较高，缺点是配置工作复杂。

(4) 被屏蔽子网。由两个包过滤路由器和一个应用网关组成。优点是：入侵者必须突破 3 个不同的设备才能侵袭内部网络；由于外部路由器只能向 Internet 通告 DMZ 网络的存在，Internet 上的系统不需要有路由器与内部网络相对；内部网络上的用户必须通过驻留在堡垒主机上的代理服务才能访问 Internet 等。

10.6.2 典型例题分析

例 1 网络系统中，通常把__(7)__置于 DMZ 区。(2014 年下半年试题 7)

(7) A．网络管理服务器　　B．Web 服务器
C．入侵检测服务器　　D．财务管理服务器

解析：DMZ 是为了解决安装防火墙后外部网络不能访问内部网络服务器的问题，而设立的一个非安全系统与安全系统之间的缓冲区，这个缓冲区位于企业内部网络和外部网络之间的小网络区域内，在这个小网络区域内可以放置一些必须公开的服务器设施，如企业 Web 服务器、FTP 服务器和论坛等。

答案：B

例 2 网络的可用性是指__(68)__。(2012 年上半年试题 68)

(68) A. 网络通信能力的大小
B. 用户用于网络维修的时间
C. 网络的可靠性
D. 用户可利用网络时间的百分比

解析：可用性不同于可靠性，可靠性是指在规定的条件下和指定的时间里，网络系统完成规定功能的能力。负载是指网络系统所能承受的数据通信量。网络的利用率太高，则设备的负荷就会过重，传输延时也要增大，衰减加快，容易出现故障。利用率低，则会造成一定程度的浪费。

答案：D

例 3　利用__(7)__可以获取某 FTP 服务器中是否存在可写目录的信息。(2011 年下半年试题 7)

(7) A. 防火墙系统　　　　　　　　B. 漏洞扫描系统
　　C. 入侵检测系统　　　　　　　D. 病毒防御系统

解析：漏洞扫描系统是用来自动检测远程或本地主机安全漏洞的程序。安全漏洞通常指硬件、软件、协议的具体实现或系统安全策略方面存在的安全缺陷。FTP 服务器中存在可写目录的信息是一个安全漏洞，可以被黑客利用，利用漏洞扫描系统可以获取某 FTP 服务器中是否存在可写目录的信息。

防火墙指的是一个由软件和硬件设备组合而成、在内部网和外部网之间、专用网与公共网之间的界面上构造的保护屏障，保护内部网免受非法用户的侵入。

入侵检测系统(IDS)是一种对网络传输进行即时监视，在发现可疑传输时发出警报或者采取主动反应措施的网络安全设备。

病毒防御系统是一个计算机防止黑客、病毒、木马的防御系统。病毒防御系统一般包括一个杀毒软件、一个病毒防火墙、一个木马黑客专杀和一个病毒监控。

答案：B

例 4　以下关于拒绝服务攻击的叙述中，不正确的是__(8)__。(2014 年下半年试题 8)

(8)　A．拒绝服务攻击的目的是使计算机或者网络无法提供正常的服务
　　B．拒绝服务攻击是不断向计算机发起请求来实现的
　　C．拒绝服务攻击会造成用户密码的泄露
　　D．DDOS 是一种拒绝服务攻击形式

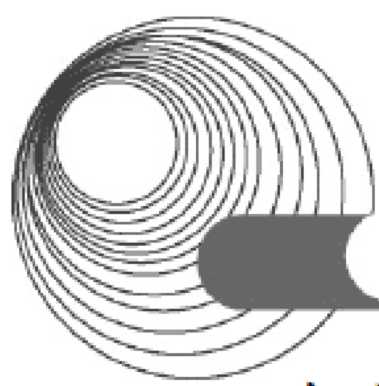
解析：拒绝服务攻击即攻击者想办法让目标机器停止提供服务，是黑客常用的攻击手段之一。其实对网络带宽进行的消耗性攻击只是拒绝服务攻击的一小部分，只要能够对目标造成麻烦，使某些服务被暂停甚至主机死机，都属于拒绝服务攻击。拒绝服务攻击问题也一直得不到合理的解决，究其原因是因为这是由于网络协议本身的安全缺陷造成的，从而拒绝服务攻击也成为了攻击者的终极手法。攻击者进行拒绝服务攻击，实际上让服务器实现两种效果：一是迫使服务器的缓冲区满，不接收新的请求；二是使用 IP 欺骗，迫使服务器把合法用户的连接复位，影响合法用户的连接。

答案：C

例 5　ARP 攻击造成网络无法跨网段通信的原因是__(8)__。(2010 年下半年试题 8)

(8) A．发送大量 ARP 报文造成网络拥塞
　　B．伪造网关 ARP 报文使得数据包无法发送到网关
　　C．ARP 攻击破坏了网络的物理连通性
　　D．ARP 攻击破坏了网关设备

解析：入侵者接收到主机发送的 ARP Request 广播包，能够偷到其他节点的(IP、MAC)地址，然后便把自己主机的 IP 地址改为合法的目的主机的 IP 地址，伪装成目的主机，然后发送一个 ping 给源主机，要求更新主机的 ARP 转换表，主机便在 ARP 表中加入新的 IP-MAC 对应关系，合法主机就失效了，入侵主机的 MAC 地址变成了合法的 MAC 地址。题目中 ARP 攻击造成网络无法跨网段通信的原因是入侵者把自己的 IP 地址改为了网关的 IP 地址，



并使得主机更新了 IP-MAC 地址对应关系，主机发送的报文则被入侵者截获，无法到达网关。

答案：B

例 6 下列选项中，防范网络监听最有效的方法是 (9) 。(2010 年下半年试题 9)

(9) A. 安装防火墙　　B. 采用无线网络传输

C. 数据加密　　D. 漏洞扫描

解析：当信息以明文形式在网络上传输时，监听并不是一件难事，只要将所使用的网络端口设置成(镜像)监听模式，便可以源源不断地截获网上传输的信息。但是，网络监听是很难被发现的，因为运行网络监听的主机只是被动地接收在局域网上传输的信息，不主动地与其他主机交换信息，也没有修改在网上传输的数据包。防范网络监听目前有这样几种常用的措施：从逻辑或物理上对网络分段，以交换式集线器代替共享式集线器，使用加密技术和划分虚拟局域网。

答案：C

例 7 防火墙的工作层次是决定防火墙效率及安全的主要因素，下面的叙述中正确的是 (8) 。(2014 年上半年试题 8)

(8) A. 防火墙工作层次越低，则工作效率越高，同时安全性越高

B. 防火墙工作层次越低，则工作效率越低，同时安全性越低

C. 防火墙工作层次越高，则工作效率越高，同时安全性越低

D. 防火墙工作层次越高，则工作效率越低，同时安全性越高

解析：防火墙工作层次越高，实现过程越复杂，对数据包的理解力越好，对非法包的判断能力越强，但工作效率越低； 防火墙工作层次越低，实现过程越简单，其工作效率越高，同时安全性越差。

答案：D

10.6.3 同步练习

1. 下面关于防火墙的说法，正确的是______。

A. 防火墙一般由软件以及支持该软件运行的硬件系统构成

B. 防火墙只能防止未经授权的信息发送到内网

C. 防火墙能准确地检测出攻击来自哪一台计算机

D. 防火墙的主要支撑技术是加密技术

2. 如果希望别的计算机不能通过 ping 命令测试服务器的连通情况，可以 (1) 。如果希望通过默认的 Telnet 端口连接服务器，则对防火墙配置正确的是 (2) 。

(1) A. 删除服务器中的 ping.exe 文件

B. 删除服务器中的 cmd.exe 文件

C. 关闭服务器中的 ICMP 端口

D. 关闭服务器中的 Net Logon 服务

(2) A.



B.



C.



D.



10.6.4　同步练习参考答案

1．A　　　2．(1) C　　(2) A

10.7　本 章 小 结

本章知识点在 2013 年的新大纲中稍微有些改动，删除了 Windows NT 系统及管理这个知识点。另外，还有一些描述方面的调整。

本章主要要求考生掌握计算机网络的基本知识，难度不大，但涉及的面广。

从历年的考题来看，每年都有专门考查网络基础知识的试题，但比例不大，一般考 5～6 道小题。主要考查各类网络协议和标准，其中有涉及 OSI 参考模型，还有就是与 Internet 有关的内容。

10.8　达标训练题及参考答案

10.8.1　达标训练题

1．国际标准化组织(ISO)的开放系统互联(OSI)参考模型共分 7 层。其中，处理系统之间用户信息的语法表达形式问题的是 (1) 层；规定通信双方相互连接的机械、电气、功能

和规程特性的是 (2) 层；向用户提供各种直接服务，如文件传送、电子邮件、虚拟终端等的是 (3) 层；通过校验和反馈重发等方法将原始不可靠的物理连接改造成无差错的数据通道的是 (4) 层；负责通信子网中从源到目标路径选择的是 (5) 层。

(1)～(5) A．物理层　B．数据链路层　C．网络层　D．运输层
E．会话层　F．表示层　G．应用层

2. 计算机的发展已进入了网络计算的新时代。Internet 是目前世界范围内最大的互联网。如此多的各种计算机之所以能通过 Internet 相互通信，是因为它们遵循了一套共同的 Internet 协议。这套协议的核心是 (1) ，在其上建立的无连接的运输层协议是 (2) ，万维网(WWW)上超文本传输遵循 (3) ，电子邮件传输遵循 (4) ，Ethernet 与 Internet 连接时要用(5)。

(1)～(5) A．TCP　B．IP　C．EDIFACT　D．HDLC　E．ARP
F．UDP　G．FTP　H．ICMP　I．SMTP　J．HTTP

3. 电子商务交易必须具备抗抵赖性，目的在于防止_____。

A．一个实体假装成另一个实体
B．参与此交易的一方否认曾经发生过此次交易
C．他人对数据进行非授权的修改、破坏
D．信息从被监视的通信过程中泄露出去

10.8.2 参考答案

1．(1) F　(2) A　(3) G　(4) B　(5) C
2．(1) B　(2) F　(3) J　(4) I　(5) E
3．B

第 11 章　标准化和软件知识产权基础知识

大纲要求：

- 标准化知识，包括标准化意识、标准化的发展、标准制订过程，国际标准、国家标准、行业标准、企业标准基本知识，代码标准、文件格式标准、安全标准、软件开发规范和文档标准知识分类，标准化机构。
- 信息化基础知识，包括信息化意识，全球信息化趋势、国家信息化战略、企业信息化战略和策略，有关的法律、法规，远程教育、电子商务、电子政务等基础知识，企业信息资源管理基础知识。

11.1　标准化基础知识

11.1.1　考点辅导

11.1.1.1　标准化的基本概念

1. 标准、标准化的概念

标准是对重复性事物或概念所做的统一规定。其主要形式有规范和规程。

标准化是在经济、技术、科学及管理等社会实践中，以改进产品、过程和服务的适用性，防止贸易壁垒，促进技术合作，促进最大社会效益为目的，对重复性事物和概念通过制定、发布和实施标准达到统一，获得最佳秩序和社会效益的过程。

2. 标准化的范围和对象

标准化的范围包括生产、经济、技术、科学及管理等社会实践中具有重复性的事物和概念以及需要建立统一技术要求的各个领域。《标准化法》规定，对下列需要统一的技术要求，应当制定标准。

(1) 工业产品的品种、规格、质量、等级或者安全、卫生要求。

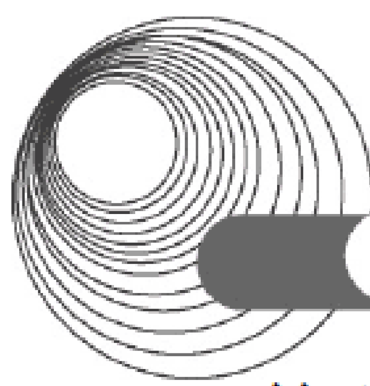
(2) 工业产品的设计、生产、试验、检验、包装、储存、运输、使用的方法或者生产、储存、运输过程中的安全、卫生要求。

(3) 有关环境保护的各项技术要求和检验方法。

(4) 建设工程的勘察、设计、施工、验收的技术要求和方法。

(5) 有关工业生产、工程建设和环境保护的技术术语、符号、代号、制图方法、互换配合要求。

(6) 农业产品的品种、规格、质量、等级、检验、包装、储存、运输以及生产技术、



管理技术的要求。

(7) 信息、能源、资源、交通运输的技术要求。

在以上领域中，凡是具有多次重复使用特性和需要制定标准的具体产品，以及各种规划、要求、方法、概念等，都可以称为标准化的对象。

标准化对象分为两大类：一类是标准化的具体对象，即需要制定标准的具体事物；另一类是标准化的总体对象，即各种具体对象的全体所构成的整体。

在企业范围内，企业的经济活动、技术活动、科研活动和管理活动的各项过程及其要素都可成为标准化的范围和对象。

3. 标准化过程模式

标准是标准化活动的产物，其目的和作用都是通过制定和贯彻具体的标准来体现的。标准化不是一个孤立的事物，而是一个活动过程。标准化活动过程一般包括标准产生(调查、研究、形成草案、批准发布)子过程、标准实施(宣传、普及、监督、咨询)子过程和标准更新(复审、废止或修订)子过程等。

1) 标准的制定

制定标准的过程，实质就是对人类社会实践经验的总结规范，每一个新标准的产生，都标志着某一领域或某项活动的经验被规范化。标准的产生过程一般包括调查研究、制订计划、起草标准、征求意见、审查、批准发布等阶段。ISO 和 IEC 是两个国际标准化组织，为规范国际标准的产生过程，发布了指导性文件，指导各国制定相应的国家标准。我国国家标准由国务院标准化行政主管部门编制计划，组织草拟，统一审批、编号、发布。

2) 标准的实施

标准的实施过程，实质就是推广和普及已被规范化的实践经验的过程。一般包括标准的宣传、贯彻执行和监督检查等。标准化活动是一项有组织的活动，通常，国家、区域或行业的标准化管理组织以及标准化团体，通过宏观管理，健全以国家标准、区域性标准、行业标准为主的标准文献资料，建立健全行业和企业标准备案管理制度，提高生产、经营和服务单位执行标准的自觉性。我国强制性标准的实施是通过强制性的监督检查来推动，依法开展标准的实施与监督。《标准化法》《国家标准管理办法》等法规，规定了我国标准化工作的方针、政策、任务和标准化体制等，它们是我国推行标准化、实施标准化管理和监督的重要依据。

3) 标准的更新

标准的更新是实践经验的深化和提高的过程。通过信息反馈总结经验和问题，依据客观环境的新变化和新要求，提出标准修订的新目标，更新标准。

4. 标准的分类

可根据不同的角度对标准进行分类。

1) 根据适用范围分类

根据标准制定的机构和标准适用的范围，可分为国际标准、国家标准、区域标准、行业标准、企业(机构)标准及项目(课题)标准。

(1) 国际标准是指国际标准化组织(ISO)、国际电工委员会(IEC)所制定的标准以及 ISO 出版的《国际标准题内关键词索引(KWIC Index)》中收录的其他国际组织制定的标准。

(2) 国家标准是由政府或国家级的机构制定或批准的、适用于全国范围的标准，是一个国家标准体系的主体和基础，国内各级标准必须服从且不得与之相抵触。例如，中华人民共和国国家标准 GB、美国国家标准 ANSI、英国国家标准 BS、日本工业标准 JIS。

(3) 区域标准泛指世界上按地理、经济或政治划分的某一区域标准化团体所通过的标准。主要有太平洋地区标准会议(PASC)、欧洲标准化委员会(CEN)、亚洲标准咨询委员会(ASAC)、非洲地区标准化组织(ARSO)等。

(4) 行业标准是由行业机构、学术团体或国防机构制定，并适用于某个业务领域的标准。例如，美国电气和电子工程师学会标准 IEEE、中华人民共和国国家军用标准 GJB、美国国防部标准 DOO-STD。

(5) 企业标准是由企业或公司批准、发布的标准，某些产品标准由其上级主管机构批准、发布。

(6) 项目标准是由某一科研生产项目组织规定，且为该项任务专用的软件工程规范，如计算机集成制造系统(CIMS)的软件工程规范。

我国标准分为国家标准、行业标准、地方标准和企业标准四级。对需要在全国范畴内统一的技术要求，应当制定国家标准。对没有国家标准而又需要在全国某个行业范围内统一的技术要求，可以制定行业标准。对没有国家标准和行业标准而又需要在省、自治区、直辖市范围内统一的工业产品的安全、卫生要求，可以制定地方标准。

企业生产的产品没有国家标准、行业标准和地方标准的，应当制定相应的企业标准，作为组织生产的依据。企业标准由企业组织制定，并按省、自治区、直辖市人民政府的规定备案。对已有国家标准、行业标准或者地方标准的，鼓励企业制定严于国家标准、行业标准或者地方标准要求的企业标准，在企业内部适用。

2) 根据标准的性质分类

根据标准的性质，可分为技术标准、管理标准和工作标准。

(1) 技术标准。技术标准主要是针对重复性的技术活动而制定的标准，是从事生产、建设及商品流通时需要共同遵守的一种技术依据。

(2) 管理标准。管理标准是管理机构为行使其管理职能而制定的具有特定功能的标准，主要用于规定人们在生产活动和社会实践中的组织结构、职责权限、过程方法、程序文件、资源分配以及方针、目标、措施、影响管理的因素等事宜，是合理组织国民经济、正确处理各种生产关系、正确实现合理分配、提高生产效率和效益的依据。分为技术管理标准、生产组织标准、经济管理标准、行政管理标准和业务管理标准等。

(3) 工作标准。工作标准是对工作的内容、方法、程序和质量要求所制定的标准。工作标准的内容主要有各岗位的职责和任务，每项任务的数量、质量要求以及完成期限，完成各项任务的程序和方法，相关岗位的协调、信息传递方式以及工作人员的考核与奖罚方法等。对生产和业务处理的先后顺序、内容和要达到的要求所作的规定称为工作工序标准。以管理工作为对象所制定的标准，称为管理工作标准。

3) 根据标准化的对象和作用分类

根据标准化的对象和作用，可分为基础标准、产品标准、方法标准、安全标准、卫生标准、环境保护标准和服务标准。

(1) 基础标准。它在一定范围内作为其他标准的基础并普遍使用，是一种具有广泛指

导意义的标准，如名词、术语、符号、代号、标识、方法、模数、公差与配合、基本参数系列、产品系列型号、产品环境条件、可靠性要求等。

(2) 产品标准。这是一定时期和一定范围内具有约束力的产品技术准则。产品标准的主要内容包括产品的适用范围，产品的品种、规格和结构形式，产品的主要性能，产品的试验、检验方法和验收规则，以及产品的包装、储存和运输等方面的要求。产品标准可分为完全的和不完全的(品种)标准。

(3) 方法标准。以各种方法为对象而制定的标准。一般包括两类：一类是以试验、检查分析、抽样、统计、计算、测定、作业等方法为对象制定的标准；另一类是为合理生产优质产品，并在生产、作业、试验、业务处理等方面为提高效率而制定的标准。

(4) 安全标准。以保护人的安全或物品的安全为对象和目的而制定的标准。一般有两种形式：一种为专门目的的安全标准；另一种是在产品标准或工艺标准中列出有关安全的要求和指标。

(5) 卫生标准。为保护人的健康，对食品、医药及其他方面的卫生要求而制定的标准。

(6) 环境保护标准。为保护环境不受污染和有利于生态平衡，对大气、水体、土壤、噪声、振动、电磁波等环境质量、污染管理、监测方法及其他事项制定的标准。

(7) 服务标准。为提高服务，对某项服务工作达到要求所制定的标准。

4) 根据法律的约束性分类

国家标准、行业标准可以分为强制性标准和推荐性标准。在我国具有法律属性，在一定范围内通过法律、行政法规等手段强制执行的标准是强制性标准；其他标准是推荐性标准。

根据《国家标准管理办法》和《行业标准管理办法》，下列标准属于强制性标准。

- 药品、食品卫生、兽药、农药和劳动卫生标准。
- 产品生产、储运和使用中的安全及劳动安全标准。
- 工程建设的质量、安全、卫生等标准。
- 环境保护和环境质量方面的标准。
- 有关国计民生方面的重要产品标准等。

推荐性标准又称为非强制性标准或自愿性标准，是指生产、交换、使用等方面，通过经济手段或市场调节而自愿采用的一类标准。这类标准，不具有强制性，任何单位均有权决定是否采用，违反这类标准不构成经济或法律方面的责任。应当指出的是，推荐性标准一经接受并采用，或各方商定同意纳入经济合同中，就成为各方必须共同遵守的技术依据，具有法律上的约束性。

国家标准(GB)中的 T 是推荐的意思。例如，GB/T 13387—1992 为推荐性标准。T 的读音为汉语拼音中的 tui。

5. 标准的代号和编号

1) ISO 的代号和编号

ISO 的代号和编号的格式：ISO+标准号+[杠+分类号] +冒号+发布年号(方括号内的内容可有可无)。

2) 国家标准的代号和编号

强制性国家标准代号为 GB，推荐性国家标准的代号为 GB/T。

国家标准的编号由国家标准的代号、标准发布顺序号和标准发布年代号组成。

(1) 强制性国家标准：GB×××××—×××××。

(2) 推荐性国家标准：GB/T×××××—×××××。

3) 行业标准的代号和编号

行业标准代号由国家主管部门审查批准公布，已公布的有 QJ(航天)、SJ(电子)、JB(机械)、JR(金融系统)等。

行业标准编号由行业标准代号、标准发布顺序及标准发布年代号组成。

(1) 强制性行业标准：×× ×××××—×××××。

(2) 推荐性行业标准：××/T ×××××—×××××。

4) 地方标准的代号和编号

地方标准的代号：由大写字母 DB 加上省、自治区、直辖市行政区划代码的前两位数字，再加上“/T”组成推荐性地方标准，不加“/T”的为强制性标准。

地方标准的编号：由地方标准代号、地方标准发布顺序号、标准发布年代号组成，表示方法如下。

(1) 强制性地方标准：DB×× ×××—××××。

(2) 推荐性地方标准：DB××/T ×××—××××。

5) 企业标准的代号和编号

企业标准的代号由大写字母 Q 加斜线“/”再加企业代号组成。

企业标准的编号由企业标准代号、标准发布顺序号和标准发布年代号组成，表示方法为“Q/××× ××××—××××。

6) 国际标准和国外先进标准

(1) 国际标准。

国际标准是指国际标准化组织(ISO)、国际电工委员会(IEC)所制定的标准，以及 ISO 出版的《国际标准题内关键词索引(KWIC Index)》中收录的其他国际组织制定的标准。

(2) 国外先进标准。

国外先进标准是指国际上有权威的区域性标准、世界上经济发达国家的国家标准和通行的团体标准，主要有下述几种。

① 有国际权威的区域性标准，如欧洲标准化委员会(CEN)、欧洲电工标准化委员会(CENELEC)、欧洲广播联盟(EBU)、亚洲大洋洲开放系统互连研讨会(AOW)、亚洲电子数据交换理事会(ASEB)等制定的标准。

② 世界经济技术发达国家的国家标准，如美国国家标准(ANSI)、德国国家标准(DIN)、英国国家标准(BS)、日本国家标准(JIS)。

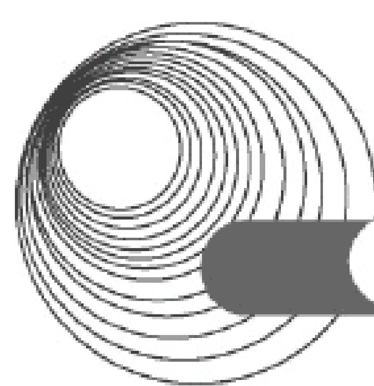
③ 国际公认的行业性团体标准，如美国材料与实验协会标准(ASTM)、美国石油协会标准(API)、美国军用标准(MIL)等。

④ 国际公认的先进企业标准，如美国 IBM 公司、美国 HP 公司、芬兰诺基亚公司等。

(3) 采用国际标准和国外先进标准的原则。

采用国际标准和国外先进标准应遵循以下原则。

① 根据我国国民经济发展的需要，确定一定时期采用国际标准和国外先进标准的方向及任务。



② 很多国际标准是经多年实践后公认的，通常不必都去进行实践验证。

③ 促进产品质量水平的提高是当前采用国际标准和国外先进标准的一项重要原则。

④ 要紧密结合我国实际情况、自然资源和自然条件，做到技术先进、经济合理、安全可靠、方便使用、促进生产力发展。

⑤ 对于国际标准中的基础标准、方法标准、原材料标准和通用零部件标准，需要先行采用。

⑥ 在技术引进和设备进口中采用国际标准，应符合《技术引进和设备进口标准化审查管理办法(试行)》中的规定。

⑦ 当国际标准不能满足要求，或尚无国际标准时，应参照以上原则，积极采用国外先进标准。

(4) 采用国际标准或国外先进标准的程度。

采用国际标准或国外先进标准的程度，分为 3 种，具体如下。

① 等同采用：指国家标准等同于国际标准，仅有或没有编辑性修改。

② 等效采用：指国家标准等效于国际标准，技术内容上只有很小差异。

③ 非等效采用：指国家标准不等效于国际标准，技术内容上有重大差异。通常包括 3 种情况：国家标准包含的内容比国际标准少；国家标准包含的内容比国际标准多；国家标准与国际标准有重叠——部分相同，但其他部分互不包括。

采用程度符号：等同为 IDT 或 idt，等效为 EQV 或 eqv，非等效为 NEQ 或 neq。

11.1.1.2 信息技术标准化

1. 信息编码标准化

编码是一种信息交换的技术手段。对信息进行编码实际上是对文字、音频、图形、图像等信息进行处理，使之量化，从而便于进行信息处理。作为一种信息交换的技术手段，必须保证信息交换的一致性。为了统一编码系统，人们借助标准化这个工具，制定各种标准代码，如国际上比较通用的 ASCII 码(美国信息交换标准代码)。

2. 汉字编码标准化

汉字编码是对每一个汉字按照一定的规律用字母、数字、符号等表示出来。汉字编码的方法很多，主要有数字编码，如电报码；拼音编码，即用汉字的拼音字母进行编码；字形编码，即用汉字的偏旁部首和笔画结构进行编码。我国在汉字编码标准化方面取得的突出成就是信息交换用汉字编码字符集国家标准的制定。

3. 软件工程标准化

软件工程的目的是改善软件开发的组织，降低开发成本，缩短开发时间，提高软件质量。软件工程最显著的特点就是把个别的、自发的、分散的软件开发变成为一种社会化的软件生产方式。软件生产的社会化必然要求软件工程实行标准化。软件工程标准的类型也是多方面的，常常是跨越软件生存期各个阶段。现已得到国家批准的软件工程国家标准有以下几类。

1) 基础标准

信息处理——《程序构造及其表示法的约定》(GB/T 13502—1992)。

信息处理系统——《计算机系统配置图符号及其约定》(GB/T 14082—1993)。
《软件工程术语标准》(GB/T 11457—1989)。
《软件工程标准分类法》(GB/T 15538—1995)。

2)　开发标准

《软件开发规范》(GB 8566—1988)。
《计算机软件单元测试》(GB/T 15532—1995)。
《软件维护指南》(GB/T 14079—1993)。

3)　文档标准

《计算机软件产品开发文件编制指南》(GB 8567—1988)。
《计算机软件需求说明编制指南》(GB/T 9385—1988)。
《计算机软件测试文件编制指南》(GB/T 9386—1988)。

4)　管理标准

《计算机软件配置管理计划规范》(GB/T 12505—1990)。
《计算机软件质量保证计划规范》(GB/T 12504—1990)。
《计算机软件可靠性和可维护性管理》(GB/T 14394—2008)(前一版本为 1993 年版)。
《信息技术、软件产品评价、质量特性及其使用指南》(GB/T 16260—1996)。

11.1.1.3　标准化组织

1. 国际标准化组织

1)　国际标准化组织

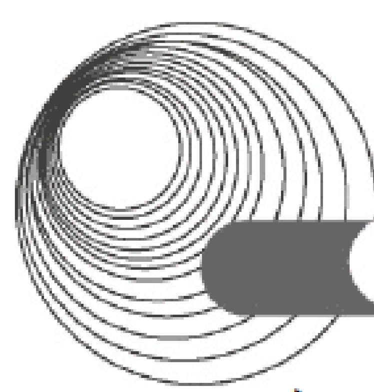
国际标准化组织(International Organization for Standardization，ISO)是目前世界上最大、最有权威性的国际标准化专门机构。1946 年 10 月 14—26 日，中、英、美、法、苏等 25 个国家的 64 名代表集会于伦敦，正式表决通过建立国际标准化组织。1947 年 2 月 23 日，ISO 章程得到 15 个国家标准化机构的认可，国际标准化组织宣告正式成立。参加 1946 年 10 月 14 日伦敦会议的 25 个国家，为 ISO 的创始人。ISO 是联合国经社理事会的甲级咨询组织和贸发理事会综合级(即最高级)咨询组织。此外，ISO 还与 600 多个国际组织保持着协作关系。

国际标准化组织的目的和宗旨是在全世界范围内促进标准化工作的发展，以便于国际物资交流和服务，并扩大在知识、科学、技术和经济方面的合作。其主要活动是制定国际标准，协调世界范围的标准化工作，组织各成员国和技术委员会进行情报交流，以及与其他国际组织进行合作，共同研究有关标准化问题。

按照 ISO 章程，其成员分为团体成员和通信成员。团体成员是指最有代表性的全国标准化机构，且每一个国家只能有一个机构代表其国家参加 ISO。通信成员是指尚未建立全国标准化机构的发展中国家或地区。通信成员不参加 ISO 技术工作，但可了解 ISO 的工作进展情况，经过若干年后，待条件成熟，可转为团体成员。ISO 的工作语言是英语、法语和俄语，总部设在瑞士日内瓦。ISO 现有成员 138 个、技术委员会(TC)187 个和分技术委员会(SC)552 个。截至 2000 年 12 月底，ISO 已制定了 13025 个国际标准。

2)　国际电工委员会

国际电工委员会(International Electrotechnical Commission，IEC)成立于 1906 年，至今已有 90 多年的历史。它是世界上成立最早的国际性电工标准化机构，负责有关电气工程和



电子工程领域中的国际标准化工作。

IEC 的宗旨是，促进电气、电子工程领域中标准化及有关问题的国际合作，增进国际间的相互了解。为实现这一目的，IEC 出版包括国际标准在内的各种出版物，并希望各成员在本国条件允许的情况下，在本国的标准化工作中使用这些标准。近 20 年来，IEC 的工作领域和组织规模均有了相当大的发展。现在 IEC 成员国已从 1960 年的 35 个增加到 60 个。他们拥有世界人口的 80%，消耗的电能占全球消耗量的 95%。目前 IEC 的工作领域已由单纯研究电气设备、电机的名词术语和功率等问题扩展到电子、电力、微电子及其应用、通信、视听、机器人、信息技术、新型医疗器械和核仪表等电工技术的各个方面。IEC 标准已涉及世界市场中的 35%的产品，到 21 世纪末，这个数字可达 50%。

IEC 标准的权威性是世界公认的。IEC 每年要在世界各地召开 100 多次国际标准会议，世界各国的近 10 万名专家在参与 IEC 的标准制订、修订工作。IEC 现在有技术委员会(TC)89 个、分技术委员会(SC)107 个。IEC 标准在迅速增加，1963 年只有 120 个标准，截至 2000 年 12 月底，IEC 已制定了 4885 个国际标准。

2. 区域标准化组织

区域标准化组织是指同处一个地区的某些国家组成的标准化组织。区域是指世界上按地理、经济或民族利益划分的区域。主要的组织有以下几个。

(1) 欧洲标准化委员会(CEN)成立于 1961 年，由欧洲共同体(EEC)、欧洲自由联盟(EFTA)所属国家的标准化机构组成，主要任务是协调各成员国的标准，制定必要的欧洲标准(EN)，实行区域认证制度。

(2) 欧洲电工标准化委员会(CENELEC)成立于 1972 年，由欧洲电工标准协调委员会(CEN EL)和欧洲电工协调委员会共同市场小组(CEN EL COM)合并组成，主要是协调各成员国电器和电子领域的标准，以及电子元器件质量认证，制定部分欧洲标准(EN)。

(3) 亚洲标准咨询委员会(ASAC)成立于 1967 年，由联合国亚洲与太平洋经社委员会协商建立，主要是在 ISO 和 IEC 标准的基础上，协调各成员国的标准化活动，制定区域性标准。

(4) 国际电信联盟 (International Telecommunications Union，ITU)于 1865 年在巴黎成立，1947 年成为联合国的专门机构，是世界各国政府的电信主管部门之间协调电信事务的一个国际组织。该组织研究制定有关电信业务的规章制度，通过决议提出推荐标准，收集有关情报。ITU 的目的和任务是维护和发展国际合作，以改进和合理利用电信，促进技术设施的发展及有效应用，提高电信业务的效率。

3. 行业标准化组织

行业标准化组织是指制定和公布适用于某个业务领域标准的专业标准化团体，以及在其业务领域开展标准化工作的行业机构、学术团体或国防机构。

(1) 美国电气电子工程师学会(Institute of Electrical and Electronic Engineers，IEEE)是美国规模最大的专业学会，由计算机和工程学专业人士组成。IEEE 主要制定的标准内容有电气与电子设备、试验方法、元器件、符号、定义以及测试方法等。近年该学会专门成立了软件标准分技术委员会(SESS)，积极开展了软件标准化活动，取得了显著成果。IEEE 通过的标准通常要报请 ANSI 审批，使之具有国家标准的性质，因此，IEEE 公布的标准常冠有 ANSI 字头，如 ANSI/IEEEStr828-1983 软件配置管理计划标准。IEEE 在通信领域最著名的

研究成果是 802 局域网标准。802 标准定义了总线网络和环形网络的通信协议。

(2) 美国国防部批准和颁布适用于美国军队内部使用的标准，代号为 DOD(采用公制计量单位的以 DOD 表示)和 MIL。

(3) 我国国防科学技术委员会批准和颁布适用于国防部门和军队使用的标准，代号为 GJB，如 1988 年发布实施的《军用软件开发规范》(GJB 437—1988)。

4. 国家标准化组织

国家标准化组织是指在国家范围内建立的标准化机构，以及政府确认的标准化团体，或者接受政府标准化管理机构指导并具有权威性的民间标准化团体。

(1) 美国国家标准学会(ANSI)。ANSI 是非营利性质的民间标准化团体，但它实际上已成为美国国家标准化中心。ANSI 协调并指导美国全国的标准化活动，给标准制订、研究和使用单位以帮助，提供国内外标准化情报。ANSI 本身很少制订标准，主要是将其他专业标准化机构的标准经协商后冠以 ANSI 代号，成为美国国家标准。

(2) 英国标准学会(BSI)。BSI 是世界上最早的全国性标准化机构，它是政府认可的、独立的、非营利性民间标准化团体。主要任务是：制订和修订英国标准；以学会名义，对各种标准进行登记，并颁发许可证；对外代表英国参加国际区域标准化活动。

(3) 德国标准化学会(DIN)。DIN 始建于 1917 年，是一个注册的公益性民间标准化机构，DIN 在国际上一直享有盛誉，其制定的标准在世界上许多国家被采用。

(4) 法国标准化协会(AFNOR)。AFNOR 成立于 1926 年，是一个公益性民间团体，也是一个被政府承认，为国家服务的组织。AFNOR 负责标准的制定和修订工作，并宣传、出版、发行标准，实施产品质量认证。

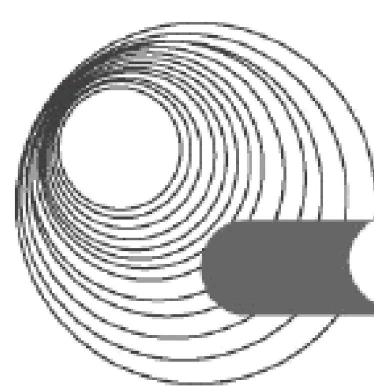
11.1.1.4　ISO 9000 标准简介

1. ISO 9000 标准

ISO 9000 标准是一族标准的统称。根据 ISO 9000-1:1994 的定义：“ISO 9000 族”是由 ISO/TC176 制定的所有国际标准。TC176 是 ISO 的第 176 个技术委员会，它成立于 1980 年，全称是“品质保证技术委员会”，1987 年更名为“品质管理和品质保证技术委员会”。TC176 专门负责制定品质管理和品质保证技术的标准。TC176 于 1986 年 6 月 15 日正式颁布了 ISO 8402《质量—术语》标准，又于 1987 年 3 月正式公布了 ISO 9000～ISO 9004 等 5 项标准，这 5 项标准和 ISO 8402:1986 一起统称为 ISO 9000 系列标准。经过全面修订，2000 年 12 月 15 日，ISO 9000: 2000 系列标准正式发布实施。ISO 9000:2000 系列标准采用了以过程为基础的质量管理体系机构模式，在标准构思和标准目的等方面体现了具有时代气息的变化，还将持续改进的思想贯穿于整个标准，把组织的质量管理体系满足顾客要求的能力和程度体现在标准的要求之中。

2. ISO 9000:2000 系列标准文件结构

ISO 9000:2000 系列标准现有 13 项标准，由 4 个核心标准、1 个支持标准、6 个技术报告、3 个小册子和 1 个技术规范构成，ISO 9000、ISO 9001、ISO 9004 和 ISO19011 这 4 项标准是 ISO 9000 族标准的核心标准。



3. ISO 9000: 2000 核心标准简介

下面介绍 ISO 9000:2000 的核心标准。

(1) ISO 9000:2000《质量管理体系——基础和术语》。该标准描述了质量管理体系的基础，并规定了质量管理体系的术语和基本原理。术语标准是讨论问题的前提；统一术语是为了明确概念，建立共同的语言。

(2) ISO 9001:2000《质量管理体系——要求》。该标准提供了质量管理体系的要求，供组织证实其具有提供满足顾客要求和适用法规要求的产品的能力时使用。该标准是用于第三方认证的唯一质量管理体系要求标准，通常用于企业建立质量管理体系以及申请认证。

(3) ISO 9004:2000《质量管理体系——业绩改进指南》。该标准给出了改进质量管理体系业绩的指南，描述了质量管理体系应包括持续改进的过程，强调通过改进过程，提高组织的业绩，使组织的顾客及其他相关方满意。

(4) ISO 19011:2000《质量管理体系和环境管理体系审核指南》。该标准提出了质量管理体系和环境管理体系审核的基本原则、审核方案的管理、环境和质量管理体系的实施以及对环境和质量管理体系评审员资格的要求。

4. ISO 9000:2000 系列标准确认的 8 项原则

下面介绍 ISO 9000:2000 系列标准确认的 8 项原则。

(1) 以顾客为中心。

(2) 领导作用。

(3) 全员参与。

(4) 过程方法。

(5) 管理的系统方法。

(6) 持续改进。

(7) 基于事实的决策方法。

(8) 互利的供求关系。

11.1.1.5 ISO/IEC 15504 过程评估标准简介

ISO/IEC 15504 是由 ISO/IEC JTC1/SC7/WG10 及其项目组 SPICE 和国际项目管理机构共同完成的标准。该标准提供了一个软件过程评估的框架，提供了一种有组织的、结构化的软件过程评估方法。在 ISO/IEC 15504 中关于过程评估的文档主要有以下几个。

(1) 概念和绪论指南。

(2) 过程和过程能力参考模型。

(3) 实施评估。

(4) 评估实施指南。

(5) 评估模型和标志指南。

(6) 评估师能力指南。

(7) 过程改进应用指南。

(8) 确定供方能力应用指南。

(9) 词汇。

11.1.2　典型例题分析

例 1　下列标准代号中，__(11)__为推荐性行业标准的代号。(2007 年下半年试题 11)

(11) A．SJ/T　　B．Q/T11　　C．GB/T　　D．DB11/T

解析：该题考查基本标准代号格式。常见的标准代号格式有以下几种。

(1) 强制性国家标准：GB ×××××—××××

(2) 推荐性国家标准：GB/T ×××××—××××

(3) 强制性行业标准编号：×× ××××—××××

(4) 推荐性行业标准编号：××/T ××××—××××

(5) 强制性地方标准编号：DB×× ×××—××××

(6) 推荐性地方标准编号：DB××/T ×××—××××

(7) 企业标准编号：Q/××× ×××—××××

从以上分析可以看出，SJ/T 为推荐性行业标准的代号，SJ/T 是电子行业的推荐性标准。

答案：A

例 2　__(10)__确定了标准体制和标准化管理体制，规定了制定标准的对象与原则以及实施标准的要求，明确了违法行为的法律责任和处罚办法。(2006 年下半年试题 10)

(10) A．标准化　　B．标准　　C．标准化法　　D．标准与标准化

解析：此题考查的是标准化相关概念，在这些概念中，标准和标准化是大家所熟知的，它们的定义如下。

标准：对重复性的事物和概念所做的统一规定，以科学、技术和实践经验的综合成果为基础，经有关方面协商一致，由一个公认机构批准，以特定形式发布，作为共同遵守的准则和依据。

标准化：是指在经济、技术、科学及管理等社会实践中，对重复性事物的概念通过制定、发布和实施标准达到统一，以获得最佳秩序和社会效益的活动。它是一门综合性学科，具有综合性、政策性和统一性的特点。

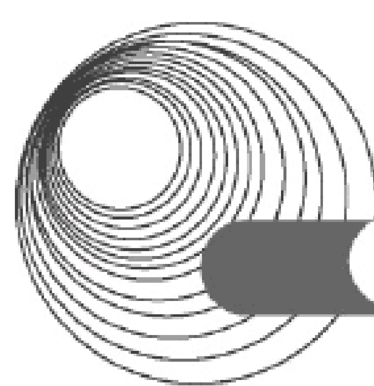
下面是对标准化法情况的简介。

标准化法即《中华人民共和国标准化法》，它由中华人民共和国第七届全国人民代表大会常务委员会第五次会议于 1988 年 12 月 29 日通过，1989 年 4 月 1 日起实施。《中华人民共和国标准化法》分为五章二十六条，其主要内容是：确定了标准体系和标准化管理体制，规定了制定标准的对象与原则以及实施标准的要求，明确了违法行为的法律责任和处罚办法。它是中华人民共和国的一项重要法律，它规定了我国标准化工作的方针、政策、任务和标准化体制等。它是国家推行标准化、实施标准化管理和监督的重要依据。

此外，标准化方面还有一个概念也是需要牢记的，即标准化过程，一般包括标准产生(调查、研究、形成草案、批准发布)、标准实施(宣传、普及、监督、咨询)和标准更新(复审、废止或修订)3 个子过程。

答案：C

例 3　上海市标准化行政主管部门制定并发布的工业产品的安全、卫生要求的标准，在其行政区域内是__(10)__。(2006 年上半年试题 10)



(10) A．强制性标准　　B．推荐性标准　　C．自愿性标准　D．指导性标准

解析：《标准化法》第二章第七条规定："国家标准、行业标准分为强制性标准和推荐性标准。保障人体健康，人身、财产安全的标准和法律、行政法规规定强制执行的标准是强制性标准，其他标准是推荐性标准。省、自治区、直辖市标准化行政主管部门制定的工业产品的安全、卫生要求的地方标准，在本行政区域内是强制性标准。"

按照我国《标准化法》的规定，上海市标准化行政主管部门制定并发布的工业产品的安全、卫生要求的标准，在其行政区域内是强制性标准。

答案：A

11.1.3　同步练习

1. 已经发布实施的标准(包括已确认或修改补充的标准)，经过实施一定时期后，对其内容再次审查，以确保其有效性、先进性和适用性，其周期一般不超过______年。

A．1　　B．3　　C．5　　D．7

2. 国家标准代号中 GB 代表中华人民共和国强制性国家标准；GB/T 代表中华人民共和国推荐性国家标准；GB/Z 代表______。

A．中华人民共和国非强制性标准
B．中华人民共和国地方标准
C．中华人民共和国行业推荐标准
D．中华人民共和国国家标准化指导性技术文件

3. X.25 建议定义了分组交换网络的接口标准，制定此标准的组织是______。

A．电子工业协会(EIA)
B．电气和电子工程师协会(IEEE)
C．国际标准化组织(ISO)
D．国际电信联盟(ITU)

4. CC 标准是 ISO 和(1)发布的一个标准。CC 是信息技术安全性通用评估准则，用来评估(2)。

(1) A．EIA　　B．IEC　　C．IEEE　　D．IETF

(2) A．网络服务的安全性
B．软件质量的安全性
C．信息系统或者信息产品的安全性
D．以上都不是

5. 在我国具有法律属性，在一定范围内通过法律、行政法规等手段强制执行的标准是强制性标准；其他标准是______。例如，GB/T 13387—1992 中的 T 就说明是这种标准。

A．地方标准　　B．推荐性标准　　C．企业标准　　D．行业标准

6. 由我国信息产业部批准发布，在信息产业部门范围内统一使用的标准，称为______。

A．地方标准　　B．部门标准　　C．行业标准　　D．企业标准

7. 《计算机软件产品开发编制指南》(GB 8657—1988)是______标准。

A．强制性国家　B．推荐性国家　　C．强制性行业　　D．推荐性行业

8. 按照标准化对象，通常把标准分为技术标准、管理标准和________三大类。
   A．工作标准　　B．信息标准　　C．服务标准　　D．国际标准

11.1.4　同步练习参考答案

1．C	2．D	3．D
4．(1) B　(2) C	5．B	6．C
7．A	8．A	

11.2　知识产权基础知识

11.2.1　考点辅导

11.2.1.1　知识产权基本概念

1．工业产权的概念

知识产权又称为智慧财产权，是指人们通过自己的智力活动创造的成果和经营管理活动中的经验、知识而依法所享有的权利。传统的知识产权可分为“工业产权”和“著作权”(版权)两类。世界贸易组织(WTO)的与贸易有关的知识产权协议(TRIPS)还把“未披露过的信息专有权”(商业秘密)、“集成电路布图设计权”列为知识产权的范围。

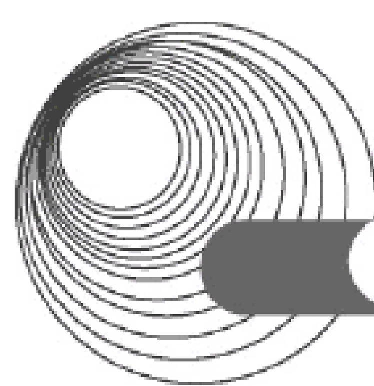
知识产权包括以下内容。

- 关于文学、艺术和科学作品的权利。
- 关于表演艺术家的表演以及唱片和广播节目的权利。
- 关于人类一切活动领域的发明的权利。
- 关于科学发现的权利。
- 关于工业品外观设计的权利。
- 关于商标、服务标记以及商业名称和标志的权利。
- 关于制止不正当竞争的权利。
- 在工业、科学、文学艺术领域内由于智力创造活动而产生的一切其他权利。

我国承认并以法律形式加以保护的主要知识产权为著作权、专利权、商标权、商业秘密以及其他有关知识产权。

1)　工业产权

工业产权包括专利、实用新型、工业品外观设计、商标、服务标记、厂商名称、产地标记或原产地名称、制止不正当竞争等项内容。此外，商业秘密、微生物技术、遗传基因技术等也属于工业产权保护的对象。发明、实用新型和工业品外观设计等属于创造性成果权利，它们都表现出比较明显的智力创造性。商标、服务标记、厂商名称、产地标记或原产地名称以及我国《反不正当竞争法》中规定的知名商品所特有的名称、包装、装潢等为识别性标记权利。



2) 著作权

著作权(又称为版权)是指作者对其创作的作品享有的人身权和财产权，包括发表权、署名权、修改权和保护作品完整权、复制权、发行权、出租权、展览权、表演权、放映权、广播权、信息网络传播权、摄制权、改编权、翻译权、汇编权、应当由著作权人享有的其他权利。著作权的保护对象包括文学、科学和艺术领域内的一切作品。

有些智力成果可以同时成为这两类知识产权保护的客体。例如，计算机软件和实用艺术品属著作权保护的同时，权利人还可以申请发明专利或外观设计专利，获得专利权，成为工业产权保护的对象。

2. 知识产权的特点

知识产权具有以下特点。

(1) 无形性。知识产权是一种无形财产权。知识产权的客体是智力创造性成果，是一种没有形体的精神财富。

(2) 双重性。某些知识产权具有财产权和人身权双重属性，如著作权。有的知识产权具有单一的属性，例如，发现权只具有名誉权属性；商业秘密只具有财产权属性，而没有人身权属性；专利权和商标权主要体现为财产权。

(3) 确认性。智力创造性成果的财产权需要依法审查确认。例如，发明人所完成的发明，其实用新型或外观设计，已经具有价值和使用价值，但是，其完成人并不能自动获得专利权，完成人必须依法提出专利申请，当获得专利局发布的授权公告后，才享有该项知识产权。文学艺术作品以及计算机软件的著作权虽然是自作品完成其权利即自动产生，但有些国家也要到登记后才能得到保护。

(4) 独占性。由于智力成果可以同时被多个主体所使用，因此，法律授予知识产权一种专有权，具有独占性。未经权利人许可使用的，就构成侵权。少数知识产权不具有独占性特征。例如，技术秘密的所有人不能禁止第三方使用其独立开发的或者合法取得的相同技术秘密，商业秘密不具备完全的财产权属性。

(5) 地域性。知识产权具有严格的地域性特点，即各国的知识产权只能在其本国领域内受法律保护。著作权虽然自动产生，但受地域限制，我国法律对外国人的作品并不都给予保护，只保护共同参加国际条约国家的公民的作品。

(6) 时间性。知识产权具有法定的保护期限，一旦保护期届满，权利将自行终止，成为社会公众可以自由使用的知识。我国的发明专利保护期为 20 年，实用新型专利权和外观设计专利权的期限为 10 年，均自专利申请日起算；我国公民的作品发表权的保护期为作者终生及其死后 50 年；我国商标权的保护期限自核准注册之日起 10 年内有效，但可申请续展注册；商业秘密权受保护的期限是不确定的，一旦该秘密为公众所知悉，即成为公众可以自由使用的知识。

3. 我国保护知识产权的法规

我国保护知识产权方面主要有以下法规。

- 《中华人民共和国著作权法》。
- 《中华人民共和国专利法》。
- 《中华人民共和国继承法》。

- 《中华人民共和国公司法》。
- 《中华人民共和国合同法》。
- 《中华人民共和国产品质量法》。
- 《中华人民共和国反不正当竞争法》。
- 《中华人民共和国刑法》。
- 《中华人民共和国计算机信息系统安全保护条例》。
- 《中华人民共和国计算机软件保护条例》。
- 《中华人民共和国著作权法实施条例》。

11.2.1.2　计算机软件著作权

1. 计算机软件著作权的主体与客体

1) 计算机软件著作权的主体

计算机软件著作权的主体指享有著作权的人，包括公民、法人和其他组织。

(1) 公民。

公民通过以下途径获得软件著作权主体资格：公民自行独立开发软件；订立委托合同，委托他人开发软件，并约定软件著作权归自己享有；通过转让途径取得软件著作财产权主体资格；公民之间或与其他主体之间，对计算机软件进行合作开发而产生的公民群体或者公民与其他主体成为计算机软件作品的著作权人；通过继承权取得的主体资格。

(2) 法人。

法人通过以下途径取得主体资格：由法人组织并提供创作物质条件所实施的开发，并由法人承担社会责任；通过接受委托、转让等各种有效合同关系而取得著作权主体资格；因计算机软件著作权主体(法人)发生变更而依法成为著作权主体。

(3) 其他组织。

其他组织指除去法人以外的能够取得计算机软件著作权的其他民事主体，包括非法人单位、合作伙伴等。

2) 计算机软件著作权的客体

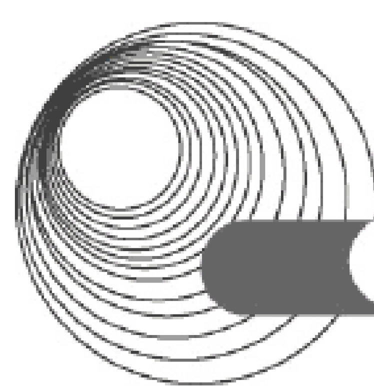
计算机软件的客体指著作权法保护的计算机软件著作权的范围，根据《著作权法》第三条和《计算机软件保护条例》第二条的规定，著作权法保护的是计算机程序及其有关文档。

(1) 计算机程序。

根据《计算机软件保护条例》第三条第一款的规定，计算机程序是指为了得到某种结果而可以由计算机等具有信息处理能力的装置执行的代码化指令序列，或者可被自动转换成代码化指令序列的符号化语句序列。计算机程序包括源程序和目标程序，同一程序的源程序文本和目标程序文本视为同一软件作品。

(2) 计算机程序的文档。

根据《计算机软件保护条例》第三条第二款的规定，计算机程序的文档是指用自然语言或者形式化语言所编写的文字资料和图表，用来描述程序的内容、组成、设计、功能规格、开发情况、测试结果及使用方法等。文档一般以程序设计说明书、流程图和用户手册等形式表现。



2. 计算机软件受著作权法保护的条件

计算机软件受著作权法保护应符合以下条件。

(1) 独立创作。受保护的软件必须由开发者独立开发创作，任何复制或抄袭他人开发的软件都不能获得著作权。软件开发的思想、概念不受著作权法的保护，如果用了他人软件作品的逻辑步骤的组合方式，则对他人软件构成侵权。

(2) 可被感知。受著作权法保护的作品是作者创作思想在固定载体上的一种实际表达。如果作者的创作思想未表达出来或不可以被感知，就不能得到著作权法的保护。因此，《计算机软件保护条例》规定，受保护的软件必须固定在某种有形物体上。

(3) 逻辑合理。受保护的计算机软件作品必须具备合理的逻辑思想，并以正确的逻辑步骤表现出来。

3. 计算机软件的著作权利

1) 计算机软件著作权的人身权

计算机软件享有两种权利，即人身权(精神权利)和财产权(经济权利)。软件著作人还享有发表权和开发者身份权。

发表权是指是否公布软件作品的权利；开发者身份权又称为署名权，指软件作者在作品中署自己名字的权利。

2) 计算机软件的著作财产权

著作财产权是指能够给著作权人带来经济利益的权利。通常是指由软件著作权人控制和支配，并能够为权利人带来一定经济效益的权利。主要内容有使用权、复制权、修改权、发行权、翻译权、注释权、信息网络传播权、出租权、使用许可权和获得报酬权、转让权。

3) 软件合法持有人的权利

软件合法持有人的权利主要有：根据使用的需要把软件装入计算机等装置内；根据需要进行必要的复制；为了防止复制品损坏而制作备份复制品；为了把该软件用于实际的计算机应用环境而做的必要修改，但不得向第三方提供修改后的软件。

4) 计算机软件著作权的行使

(1) 软件经济权利的许可使用。

软件经济权利的许可使用是指软件著作权人通过合同方式许可他人使用其软件，并获得一定报酬的软件贸易形式。主要有：独占许可使用，被授权方按合同规定取得软件使用的独占性，权利人不得将使用权授予第三方，自己也不得使用该软件；独家许可使用，权利人自己可以使用该软件，其他和独占许可使用相同；普通许可使用，权利人可以将使用权授予第三方，自己也可以使用；法定许可使用和强制许可使用，根据法律特殊规定，不经软件著作权人许可也可以使用其软件。

(2) 软件经济权利的转让使用。

软件经济权利的转让使用是指软件著作权人将其著作权中的经济权利全部转移给他人，受让者成为新的著作权主体。软件著作权的转让必须签订书面合同，同时转让不改变软件的保护期。转让方式包括卖出、赠与、抵押、赔偿等。

5) 计算机软件著作权的保护期

计算机软件著作权自软件开发完成之日起，保护期为 50 年。保护期满，除开发者身份

权外，其他权利终止。计算机软件著作权人的单位终止和计算机软件著作权人的公民死亡无合法继承人时，除开发者身份权外的其他权利进入公有领域。

4. 计算机软件著作权的归属

我国《著作权法》规定著作权属于作者。《计算机软件保护条例》规定软件著作权属于软件开发者。

1)　职务开发软件著作权的归属

公民为完成法人或者其他组织工作任务所创作的作品是职务作品，著作权由作者享有，但法人或者其他组织有权在其业务范围内优先使用。作品完成两年内，未经单位同意，作者不得许可第三人以与单位使用的相同方式使用该作品。

有下列情形之一的职务作品，作者享有署名权，著作权的其他权利由法人或者其他组织享有，法人或者其他组织可以给予作者奖励。

(1)　主要是利用法人或者其他组织的物质技术条件创作，并由法人或者其他组织承担责任的工程设计图、产品设计图、地图、计算机软件等职务作品。

(2)　法律、行政法规规定或者合同约定著作权由法人或者其他组织享有的职务作品。

2)　合作开发软件著作权的归属

合作开发软件是指两个或两个以上公民、法人或其他组织订立协议，共同参加某项计算机软件的开发并分享软件著作权的形式。对合作开发软件著作权的归属应掌握以下 4 点。

(1)　由两个以上的单位、公民共同开发完成的软件属于合作开发的软件。

(2)　由于合作开发软件著作权是由两个以上单位或者个人共同享有，因而为了避免在软件著作权的行使中产生纠纷，规定“合作开发的软件，其著作权的归属由合作开发者签订书面合同约定”。

(3)　对于合作开发的软件著作权按以下规定执行：“无书面合同或者合同未作明确约定，合作开发的软件可以分割使用的，开发者对各自开发的部分可以单独享有著作权；但是行使著作权时不得扩展到合作开发的软件整体的著作权。”

(4)　合作开发者对于软件著作权中的转让权不得单独行使。

3)　委托开发软件著作权的归属

受委托创作的作品，著作权的归属由委托人和受托人通过合同约定。合同未作明确约定或者没有订立合同的，著作权属于受托人。委托开发的软件著作权的归属按以下标准确定。

(1)　委托开发软件作品需根据委托方的要求，由委托方与受托方以合同确定的权利和义务的关系而进行开发的软件。因此，软件作品著作权归属应当作为合同的重要条款予以明确约定。

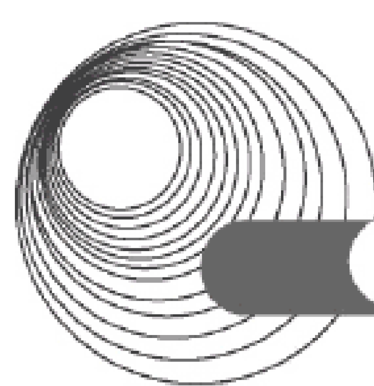
(2)　若在委托开发软件活动中，委托者与受委托者没有签订书面协议，或者在协议中未对软件著作权归属作出明确的约定，则软件著作权属于受委托者，即属于实际完成软件的开发者。

4)　接受任务开发软件的著作权归属

接受任务开发软件的著作权归属一般按以下两条标准确定：①在合同中明确约定的，按照合同约定实行；②未明确约定的，著作权属于实际完成软件开发的单位。

5)　计算机软件著作权主体变更后软件著作权的归属

因主体变更引起的变化有以下几种。



(1) 公民继承的软件权利归属。合法继承人享有除署名权外的其他权利，如著作权的使用权、使用许可权和获得报酬权等权利。

(2) 单位变更后软件权利归属。著作权属于法人或者其他组织的，法人或者其他组织变更、终止后，由承受其权利义务的法人或者其他组织享有；没有承受其权利义务的法人或者其他组织的，由国家享有。

(3) 权利转让后的软件著作权归属。权利转让根据签订的合同规定各方的权利。

(4) 司法判决、裁定引起的软件著作权归属问题根据法律的判决来执行。

(5) 保护期届满权利丧失。

5. 计算机软件著作权侵权的鉴别

1) 计算机软件著作权侵权行为

计算机软件著作权侵权行为主要有：未经软件著作权人的同意而发表或者登记其作品；将他人开发的软件当作自己的作品发表或者登记；未经合作者同意将与他人合作开发的软件当作自己独立完成的作品发表或者登记；在他人开发的软件上署名或者更改他人开发的软件上的署名；未经软件著作权人或者其合法受让者的许可，修改或翻译其软件作品；未经软件著作权人或其合法受让者的许可，复制或部分复制其软件；未经软件著作权人或其合法受让者的同意，向公众发行出租其软件的复制品；未经软件著作权人或其合法受让者的同意，向任何第三方办理软件权利许可或转让事宜；未经软件著作权人或其合法受让者的同意，通过信息网络传播著作权人的软件；共同侵权，两人以上共同实施的侵权行为。

2) 不构成计算机软件侵权的合理使用行为

根据已获得的软件使用权利进行的不超出使用权限的活动都是合法使用。区分合理使用和不合理使用可以按以下标准。

(1) 软件作品是否合法取得。

(2) 使用目的是否具有商业营业性，如果是就不属于合理使用。

(3) 合理使用一般为少量的使用，超过通常认为的少量界限，即可认为不属于合理使用。

3) 计算机著作权软件侵权的识别

计算机软件作为《著作权法》保护的客体，具有以下特点。

(1) 技术性，指其创作和开发的高技术性。

(2) 依赖性，指人们对其的了解依赖于计算机。

(3) 多样性，指计算机程序表达的多样性。

(4) 运行性，指程序功能的可运行性。

识别侵权软件可采取下列方法：①将正版和盗版软件进行比对；②将两套软件同时或先后安装，观察其显示是否相同；③对其安装后的目录和各种文件进行对比；④在使用过程中进行对比；⑤进行源程序的对比。

6. 计算机软件著作权侵权的法律责任

计算机软件著作权侵权的法律责任主要有民事责任、行政责任和刑事责任。

需要承担民事责任的侵权行为有：未经软件著作权人的许可发表或登记其软件的；将他人的软件当作自己的软件发表或登记的；未经合作者许可，将与他人合作开发的软件当作自己独立完成的作品发表或者登记的；在他人开发的软件上署名或者更改他人开发的软

件上署名的；未经软件著作权人或者其合法受让者的许可，修改或翻译其软件的；其他侵犯软件著作权的行为。

需要承担行政责任的侵权行为有：复制或部分复制著作权人软件的；向公众发行、出租著作权人软件的；故意避开或者破坏著作权人为保护其软件而采取的技术措施的；故意删除或者改变软件权利管理电子信息的；许可他人行使或者转让著作权人的软件著作权的。

侵权行为构成犯罪的，侵权者应承担相应的刑事责任。

11.2.1.3　计算机软件的商业秘密权

1. 商业秘密

1)　商业秘密的定义

在我国的《反不正当竞争法》中规定商业秘密是“不为公众所熟悉的、能为权利人带来经济效益、具有实用性并经权利人采取保密措施的技术信息和经营信息”，主要包括经营秘密和技术秘密。

商业秘密权作为一种无形财产权受到法律的保护，在计算机软件中，包含商业秘密的可作为商业秘密权的保护对象。

2)　商业秘密的构成条件

商业秘密的构成条件是：商业秘密必须具有未公开性，即不为公众所知悉；商业秘密必须具有实用性，即能为权利人带来经济效益；商业秘密必须具有保密性，即采取了保密措施。

3)　商业秘密权

商业秘密是一种无形的信息财产。与有形财产相区别，商业秘密不占据空间，不易被权利人所控制，不发生有形损耗，其权利是一种无形财产权。商业秘密的权利人与有形财产所有权人一样，依法享有占有、使用和收益的权利，即有权对商业秘密进行控制与管理，防止他人采取不正当手段获取与使用；有权依法使用自己的商业秘密，而不受他人干涉；有权通过自己使用或者许可他人使用以至转让所有权，从而取得相应的经济利益；有权处理自己的商业秘密，包括放弃占有、无偿公开、赠与或转让等。

4)　商业秘密的丧失

一项商业秘密受到法律保护的依据，是必须具备上述构成商业秘密的 3 个条件，当缺少上述 3 个条件之一时就会造成商业秘密丧失保护。

2. 计算机软件商业秘密的侵权

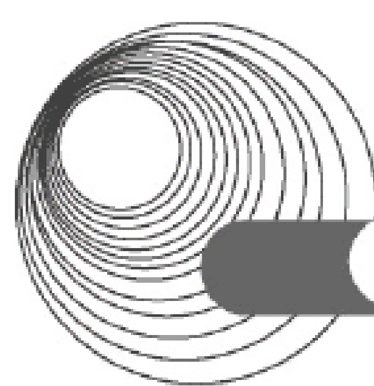
侵犯商业秘密是指未经权利人的许可，以非法手段获得商业秘密并加以公开或使用的行为。其具体行为主要有以下几种。

(1) 以盗窃、利诱、胁迫或以其他不正当手段获取权利人的计算机软件商业秘密。

(2) 披露、使用或允许他人使用以不正当手段获取的计算机软件商业秘密。

(3) 违反约定或违反权利人有关保守商业秘密的要求，披露、使用或允许他人使用其掌握的计算机软件商业秘密的。

(4) 第三方在明知前述违法行为的情况下，仍然从侵权人那里获取或使用他人计算机软件商业秘密的。该行为属于间接侵权。



3. 计算机软件商业秘密侵权的法律责任

下面介绍计算机软件商业秘密侵权的法律责任。

(1) 侵权者的行政责任。责令其停止侵权行为，处以 1 万元以上 20 万元以下的罚款。

(2) 侵权者的民事责任。侵权人承担损害赔偿责任，被侵权人可以向法院提起诉讼。

(3) 侵权者的刑事责任。侵权者的侵权行为造成重大损害的，侵权者承担刑事责任。

11.2.1.4 专利权概述

1. 专利权的保护对象与特征

专利权是指由国务院专利行政部门授予的，发明创造者在规定的时间内享有的独占使用权，在这一规定的时间内，任何自然人、法人、其他组织，未经其许可，均不得制造其发明创造。依《中华人民共和国专利法》规定，我国国务院专利行政管理部门授予的专利有以下 3 种，即发明专利、实用新型专利、外观设计专利。不属于专利权范围的有以下几种。

(1) 违反国家法律、社会公德或者妨碍公共利益的发明创造。

(2) 科学发现。

(3) 智力活动的规则和方法，如推理、分析、判断、处理等思维活动的方法。

(4) 疾病诊断手段和治疗方法。

(5) 动、植物品种。

(6) 用原子核变换方法获得的物质。

2. 授予专利的条件

授予专利的条件指发明创造获得专利的实质条件，包括新颖性、创造性和实用性 3 个方面。

(1) 新颖性是指发明创造在申请日之前未被公开也没有同样的发明被申请的。有些发明虽然被公开，但在一定期限内仍然具有新颖性。在申请日之前 6 个月内，有下列情况之一，仍有新颖性：在中国政府主办或者承认的国际展览会上首次展出的；在规定的学术会议或者技术会议上首次发表的；他人未经申请人同意而泄露其内容的。新颖性是创造性的前提。

(2) 创造性是指发明创造要有实质性特点和显著的进步。例如，解决了没有解决的技术难题；克服了技术偏见；取得了新的技术效果等。

(3) 实用性是指发明创造要能够使用，并且能够产生积极的效果。申请专利的技术方案违背自然规律或利用独一无二的自然条件完成的，不具备实用性。

外观设计获得专利权的条件为新颖性和美观性。新颖性和上述相同，美观性是指该设计可以使产品产生美感。

3. 专利的申请

专利申请权是公民、法人或其他组织依据法律规定或者合同约定享有的就发明创造向专利局提出专利申请的权利。专利申请权可以转让、继承或赠与。

专利申请人是指对某项发明创造依法律规定或者合同约定享有专利申请权的公民、法人或者其他组织，包括职务发明创造的单位、非职务发明创造的发明人或设计人、共同发明创造的发明人、委托发明创造的受让人。

专利申请采用书面形式，一项专利申请文件只能申请一项专利。两个或两个以上的人就同样的发明创造申请专利的，专利权授予最先申请人。两个以上的申请人在同一日分别就同样的发明创造申请专利的，应在收到专利行政部门的通知后自行协商确定申请人。

发明或者实用新型的专利申请文件包括请求书、说明书、说明书摘要、权利要求书。外观设计专利申请文件包括请求书、图片或照片。

专利申请日，又称关键日，是指专利主管部门收到完整的专利申请文件的日期。如果专利申请文件是邮寄的，以寄出的邮戳日为申请日。

对发明专利的审批要通过实质审查，即依法审查专利的新颖性、创造性和实用性。对实用新型和外观设计专利申请只进行初步审查，不进行实质审查。

申请人在法定期间或专利局指定的期限内未办理相关的手续或未能提供有关文件的，其申请将被撤回，丧失其申请权。

4. 专利权行使

1) 专利权的归属

依《中华人民共和国专利法》及其实施细则的规定，专利权归下列人所有。

(1) 职务发明创造的专利申请权和专利权人为单位。

(2) 非职务发明创造的专利申请权和专利权人为个人。

(3) 利用本单位的物质技术条件所完成的发明创造，其专利申请权和专利权人依其合同约定决定。

(4) 两个以上单位或者个人合作完成的发明创造，除各方在协议中约定的以外，其专利申请权和专利权人属于完成或者共同完成的单位或者个人。

(5) 一个单位或者个人接受其他单位或者个人的委托完成的发明创造，除委托书中有约定的外，其专利申请权和专利权人属于完成或者共同完成的单位或者个人。

(6) 两个以上的申请人分别就同样的发明创造申请专利的，专利权授予最先申请的人。

(7) 委托开发的专利权根据委托开发的协议中的规定来确定，若未有明确规定，则专利权属于专利完成者。

2) 专利权人的权利

专利权是一种具有财产权属性的独占权以及由其衍生出来的相应处分权。专利权人的权利有独占实施权、转让权、实施许可权、放弃权、标记权等。专利实施许可包括独占许可、独家许可、普通许可和部分许可。

5. 专利权的限制

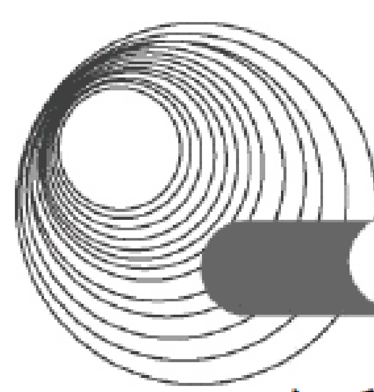
根据我国《专利法》的规定，发明专利的保护期限为 20 年，实用新型和外观设计专利为 10 年。

专利权因某种法律事实的发生而导致其效力消失的情形称为专利权终止。导致专利终止的事实有：保护期限届满；专利权人放弃专利权；专利权人没有按规定交纳年费的。

6. 专利侵权行为

专利侵权行为是指在专利保护期内，未经专利权人的许可擅自以营利为目的使用专利的行为。主要有以下几种。

(1) 为生产经营目的的制造、使用、销售其专利产品，或者使用其专利方法以及通过该



专利方法获得产品。

(2) 为生产经营目的制造、销售其外观设计专利产品。

(3) 进口依照其专利方法直接获得的产品。

(4) 产品的包装上标明专利标记和专利号。

(5) 冒充专利产品或专利方法等。

对专利侵权行为，专利权人可以请求专利管理机关处理，也可以请求法院审理。侵犯专利的诉讼时效为两年，自专利权人知道或应当知道侵权之日起算起。

11.2.1.5 企业知识产权的保护

1. 企业知识产权管理

应建立管理企业诸如专利权、商标权、著作权等资源的专门机构，以有效地管理和利用知识产权。

2. 知识产权的保护和利用

目前计算机技术和软件技术的知识产权保护以《著作权法》为主，《专利法》《商标法》《反不正当竞争法》《合同法》为辅。例如，源程序及设计文档作为软件的表现形式受《著作权法》的保护，同时作为技术秘密又受《反不正当竞争法》的保护。对企业来说，不能只依靠法律法规，要建立自己的知识产权保护措施，一般可采取以下方法。

(1) 明确软件知识产权归属。

(2) 及时对软件技术秘密采取保护措施。一旦发生泄露，要追究行为人的责任，保护企业权益。

(3) 依靠专利保护新技术和新产品。要及时申请专利，不能拖延导致新成果新颖性的丧失。

(4) 软件产品要尽快完成商标或服务标记的注册，保护产品的商标专有权。

(5) 软件产品进入市场前进行软件著作权登记。

3. 建立经济约束机制规范调整各种关系

软件企业需要建立企业内部以及企业与外部的各种经济约束机制。目前，软件企业应建立以下各项合同规范。

(1) 劳动关系合同。

(2) 软件开发合同。约定软件开发各方享有的权利和义务。

(3) 软件许可使用(或转让)合同。

11.2.2 典型例题分析

例 1 王某是一名软件设计师，按公司规定编写软件文档，并上交公司存档。这些软件文档属于职务作品，且<u> (10) </u>。(2013 年上半年试题 10)

(10) A. 其著作权由公司享有

B. 其著作权由软件设计师享有

C. 除其署名权以外，著作权的其他权利由软件设计师享有

D. 除著作权由公司和软件设计师共同享有

解析：执行本单位的任务或者主要是利用本单位的物质条件所完成的职务发明创造，著作的权利属于该单位。王某所编写的文档是按公司规定编写的，因此著作权应归属该公司。

答案：A

例 2　甲经销商擅自复制并销售乙公司开发的 OA 软件光盘已构成侵权。丙企业在未知的情形下从甲经销商处购入 10 张并已安装使用。在丙企业知道了所使用的软件为侵权复制品的情形下，以下说法正确的是 (11) 。(2013 年上半年试题 11)

(11) A. 丙企业的使用行为为侵权，须承担赔偿责任

B. 丙企业的使用行为不侵权，可以继续使用这 10 张软件光盘

C. 丙企业的使用行为侵权，支付合理费用后可以继续使用这 10 张软件光盘

D. 丙企业的使用行为不侵权，不需要承担任何法律责任

解析：根据我国《反不正当竞争法》第十条规定，侵犯计算机的具体表现形式包括第三人在明知违法的情况下，仍然从侵权人那里获取、披露或者使用的计算机软件商业秘密。因此该公司行为侵权，支付合理费用后可以继续使用这 10 张光盘。

答案：C

例 3　M 软件公司的软件产品注册商标为 M，为确保公司在市场竞争中占据优势，对员工进行了保密约束。此情形下该公司不享有 (10) 。(2012 年下半年试题 10)

(10) A．商业秘密权　B．著作权　C．专利权　D．商标权

解析：专利权不能自动取得，必须向专利行政部门提出专利申请，履行专利法规定的专利申请手续并向国家专利行政部门提交必要的申请文件。题目中，M 软件公司没有申请过专利，因此不享有专利权。

答案：C

例 4　X 软件公司的软件工程师张某兼职于 Y 公司，为完成 Y 科技公司交给的工作，作出了一项涉及计算机程序的发明。张某认为该发明是利用自己的业余时间完成的，可以以个人名义申请专利。此项专利申请权应归属于 (11) 。(2012 年下半年试题 11)

(11) A．张某　B．X 软件公司　C．Y 科技公司　D．张某和 Y 科技公司

解析：张某的发明为了完成 Y 公司交给的工作而产生的结果，很显然该发明为职务发明创造。专利权法规定，职务发明创造的专利申请和获得专利的权利为单位所有。

答案：C

例 5　软件著作权的客体不包括 (10) 。(2012 年上半年试题 10)

(10) A. 源程序　B. 目标程序　C. 软件文档　D. 软件开发思想

解析：软件著作权的客体是指著作权法保护的软件著作权的范围，包括计算机程序、软件开发文档，而计算机程序又包括源程序和目标程序。受著作权法保护的作品应当是作者创作思想在固定载体上的一种实际表达，而创作思想未表达出来，不可被感知，就得不到著作权法的保护。

答案：D

例 6　中国企业 M 与美国公司 L 进行技术合作，合同约定 M 使用一项在有效期内的美国专利，但该项美国专利未在中国和其他国家提出申请。对于 M 销售依照该专利生产的产

品，以下叙述正确的是 (11) 。(2012 年上半年试题 11)

(11) A. 在中国销售，M 需要向 L 支付专利许可使用费
B. 返销美国，M 不需要向 L 支付专利许可使用费
C. 在其他国家销售，M 需要向 L 支付专利许可使用费
D. 在中国销售，M 不需要向 L 支付专利许可使用费

解析：专利权具有合法性、独占性、地域性、时间性和经济性的特征。地域性是指一个国家依照其本国专利法授予的专利权，仅在该国法律管辖的范围内有效，对其他国家没有任何约束力，外国对其专利权不承担保护的义务。题目中涉及的专利是在美国取得的，那么专利权人只在美国享有专有权或独占权。在我国使用该专利权，不属于侵权行为，不需要支付专利许可使用费。

答案：D

例 7 (10) 指可以不经著作权人许可，不需要支付报酬，使用其作品。(2011 年下半年试题 10)

(10) A. 合理使用　　B. 许可使用
C. 强制许可使用　　D. 法定许可使用

解析：著作权的合理使用属于著作权的一种限制，它是指自然人、法人或者其他组织为了个人欣赏、评论、新闻报道、教学与学术研究以及公益事业等目的，根据著作权法的规定，可以不经过作者同意而使用其已经发表的作品，不需要向其支付报酬。合理使用的对象是已经发表的作品。合理使用应尊重作者的人身权利，应当指明作者姓名、作品名称，并且不得影响作品的正常使用，也不得不合理损害著作权人的合法利益。合理使用成立的条件都是由法律予以规定的，因此可以说合理使用也是一种特别的法定许可使用。

许可使用是指软件著作权人或权利合法受让者，通过合同方式许可他人使用其软件，并获得报酬的一种软件贸易形式。法定许可使用和强制许可使用都属于许可使用。法定许可，是由法律规定使用他人作品时，不需要征得著作权人的同意，但需要向著作权人支付报酬。强制许可是指在著作权人无正当理由而拒绝与使用人达成使用作品的协议时，被拒绝人可以向法定部门申请并获得授权使用作品，但是应当尊重著作权人的人身权利，并且支付报酬。

答案：A

例 8 王某是 M 国际运输有限公司计算机系统管理员。任职期间，王某根据公司的业务要求开发了“海运出口业务系统”，并由公司使用。随后，王某向国家版权局申请了计算机软件著作权登记，并取得了《计算机软件著作权登记证书》。证书明确软件名称是“海运出口业务系统 V1.0”，著作权人为王某。以下说法正确的是 (11) 。(2011 年下半年试题 11)

(11) A. 海运出口业务系统 V1.0 的著作权属于王某
B. 海运出口业务系统 V1.0 的著作权属于 M 公司
C. 海运出口业务系统 V1.0 的著作权属于王某和 M 公司
D. 王某获取的软件著作权登记证是不可以撤销的

解析：软件开发完成后，著作权自动产生，不论是否登记都享有著作权。软件著作权登记证书是对登记事项的初步证明，可以帮助持有者在诉讼中起到减轻举证责任的作用。

王某在单位任职期间，针对本职工作中明确指定的任务所开发的软件为职务开发软件，该软件的著作权由单位享有。因此，海运出口业务系统 V1.0 的著作权属于 M 公司。

中国版权保护中心可以根据申请人的申请，撤销软件著作权登记证。

答案：B

例 9　下列关于软件著作权中翻译权的叙述不正确的是：翻译权是指__(10)__的权利。(2011 年上半年试题 10)

(10) A．将原软件从一种自然语言文字转换成另一种自然语言文字
B．将原软件从一种程序设计语言转换成另一种程序设计语言
C．软件著作权人对其软件享有的以其他各种语言文字形式再表现
D．对软件的操作界面或者程序中涉及的语言文字翻译成另一种语言文字

解析：根据《计算机软件保护条例》第八条的规定，软件著作权人享有若干项权利，其中包括翻译权。在条例中对翻译权的定义是："将原软件从一种自然语言文字转换成另一种自然语言文字的权利。"

答案：B

例 10　某软件公司研发的财务软件产品在行业中技术领先，具有很强的市场竞争优势。为确保其软件产品的技术领先及市场竞争优势，公司采取相应的保密措施，以防止软件技术秘密的外泄。并且，还为该软件产品冠以"用友"商标，但未进行商标注册。此情况下，公司仅享有该软件产品的__(11)__。(2011 年上半年试题 11)

(11) A．软件著作权和专利权
B．商业秘密权和专利权
C．软件著作权和商业秘密权
D．软件著作权和商标权

解析：专利权不能自动取得，必须向专利行政部门提出专利申请。题目中软件没有进行专利申请，因此不享有专利权，可以排除选项 A、B。商标权是商标所有人依法对其注册商标享有的专有使用权。未注册的商标不受商标法保护。题目中软件产品使用的商标没有注册，因此不享有商标权，可以排除选项 D。商业秘密权是商业秘密的合法控制人采取保密措施依法对其经营信息和技术信息享有的专有使用权，题目中提到"公司采取相应的保密措施，以防止软件技术秘密的外泄"，所以享有商业秘密权。

答案：C

例 11　软件商标权的权利人是指__(10)__。(2010 年下半年试题 10)

(10) A．软件商标设计人　　B．软件商标制作人
C．软件商标使用人　　D．软件注册商标所有人

解析：软件商标权的权利人是指软件注册商标所有人。

答案：D

例 12　利用__(11)__可以对软件的技术信息、经营信息提供保护。(2010 年下半年试题 11)

(11) A．著作权　　B．专利权　　C．商业秘密权　　D．商标权

解析：在《反不正当竞争法》中商业秘密被定义为"不为公众所知悉的、能为权利人带来经济利益的、具有实用性并经权利人采取保密措施的技术信息和经营信息"。软件中包含着技术秘密和经营秘密，具有商业秘密的特征，即使软件尚未开发完成，在软件开发

中所形成的知识内容也构成商业秘密。因此，可以利用商业秘密权对软件的技术信息、经营信息提供保护。

答案：C

例 13 李某在某软件公司兼职，为完成该公司交给的工作，作出了一项涉及计算机程序的发明。李某认为该发明是自己利用业余时间完成的，可以个人名义申请专利。关于此项发明的专利申请权应归属 (12) 。(2010 年下半年试题 12)

(12) A．李某　　　　B．李某所在单位
C．李某兼职的软件公司　　　　D．李某和软件公司约定的一方

解析：《中华人民共和国专利法》规定，执行本单位的任务或者主要是利用本单位的物质条件所完成的职务发明创造，申请专利的权利属于该单位。李某所作的发明是其在软件公司兼职时本职工作的结果，因此专利申请权应归属软件公司。

答案：C

例 14 两个以上的申请人分别就相同内容的计算机程序的发明创造，先后向国务院专利行政部门提出申请， (10) 可以获得专利申请权。(2010 年上半年试题 10)

(10) A．所有申请人均　　　　B．先申请人
C．先使用人　　　　D．先发明人

解析：在我国，审批专利遵循的基本原则是"先申请先得"原则，即对于同样的发明创造，谁先申请专利，专利权就授予谁。专利法第九条规定，两个以上的申请人分别就同样的发明创造申请专利的，专利权授予最先申请的。当有二者在同一时间就同样的发明创造提交了专利申请，专利局将分别向各申请人通报有关情况可以将两申请人作为一件申请的共同申请人，或其中一方放弃权利并从另一方得到适当的补偿，或两件申请都不授予专利权。但专利权的授予只能给一个人。

答案：B

例 15 王某是一名程序员，每当软件开发完成后均按公司规定完成软件文档，并上交公司存档，自己没有留存。因撰写论文的需要，王某向公司要求将软件文档原本借出复印，但遭到公司拒绝，理由是该软件文档属于职务作品，著作权归公司。以下叙述正确的是 (11) 。(2010 年上半年试题 11)

(11) A．该软件文档属于职务作品，著作权归公司
B．该软件文档不属于职务作品，程序员享有著作权
C．该软件文档属于职务作品，但程序员享有复制权
D．该软件文档不属于职务作品，著作权由公司和程序员共同享有

解析：《计算机软件保护条例》第十三条作出了明确规定：公民在单位任职期间所开发的软件，如果是执行本职工作的结果，即针对本职工作供明确指定的开发目标所开发的，或者是从事本职工作活动所预见的结果或自然的结果，则该软件的著作权属于该单位。题目中，软件文档属于职务作品，著作权归公司。

答案：A

例 16 下列智力成果中，能取得专利权的是(10)。(2009 年下半年试题 10)

(10) A．计算机程序代码　　　　B．游戏的规则和方法
C．计算机算法　　　　D．用于控制测试过程的程序

解析：授予专利权的条件是指一项发明创造获得专利权应当具备的实质性条件。一项

发明或者实用新型获得专利权的实质条件为新颖性、创造性和实用性。因此容易得出答案为 C。

答案：C

例 17　软件权利人与被许可方签订一份软件使用许可合同。若在该合同约定的时间和地域范围内，软件权利人不得再许可任何第三人以此相同的方法使用该项软件，但软件权利人可以自己使用，则该项许可使用是(11)。(2009 年下半年试题 11)

(11) A．独家许可使用　　B．独占许可使用
C．普通许可使用　　D．部分许可使用

解析：本题考查软件经济权利的许可使用。

软件经济权利的许可使用是指软件著作权人或权利合法受让者，通过合同方式许可他人使用其软件，并获得报酬的一种软件贸易形式。许可使用的方式可分为以下几种。

(1) 独占许可使用。权利人通过书面合同授权，被授权方可以根据合同规定的方式、条件和时间确定独占性，权利人不得将软件使用权授予第三方，权利人自己不能使用该软件。

(2) 独家许可使用。权利人通过书面合同授权，被授权方可以根据合同规定的方式、条件和时间确定独占性，权利人不得将软件使用权授予第三方，权利人自己可以使用该软件。

(3) 普通许可使用。权利人通过书面合同授权，被授权方可以根据合同规定的方式、条件和时间确定独占性，权利人可以将软件使用权授予第三方，权利人自己可以使用该软件。

(4) 法定许可使用和强制许可使用。在法律特定的条款下，不经软件著作权人许可，使用其软件。

答案：A

例 18　关于软件著作权产生的时间，以下表述正确的是(10)。(2009 年上半年试题 10)

(10) A．自作品首次公开发表时
B．自作者有创作意图时
C．自作品得到国家著作权行政管理部门认可时
D．自作品完成创作之日

解析：依《计算机软件保护条例》第十四条相关规定，软件著作权自软件开发完成之日起产生。

答案：D

例 19　甲、乙两公司的软件设计师分别完成了相同的计算机程序发明，甲公司先于乙公司完成，乙公司先于甲公司使用。甲、乙公司于同一天向专利局申请发明专利。此情形下，(11)可获得专利权。(2015 年上半年试题 11)

(11) A．甲公司　　B．甲、乙公司均
C．乙公司　　D．由甲、乙公司协商确定

解析：专利权的申请原则为：谁先申请谁获得，同时申请协商确定。

答案：(11) D

例 20　某软件公司参与开发管理系统软件的程序员张某，辞职到另一公司任职，于是该项目负责人将该管理系统软件上开发者的署名更改为李某(接替张某工作)。该项目负责人的行为(10)。(2016 年上半年试题 10)

(10) A．侵犯了张某开发者身份权(署名权)
B．不构成侵权，因为程序员张某不是软件著作权人

C．只是行使管理者的权利，不构成侵权

D．不构成侵权，因为程序员张某现已不是项目组成员

解析：根据我国《著作权法》第九条和《计算机软件保护条例》第八条的规定，软件著作权人享有发表权和开发者身份权，这两项权利与著作权人的人身是不可分离的主体。其中，开发者的身份权，不随软件开发者的消亡而丧失，且无时间限制。张某参加某软件公司开发管理系统软件的工作，属于职务行为，该管理系统软件的著作权归属公司所有，但张某拥有该管理系统软件的署名权。而该项目负责人将作为软件系统开发者之一的张某的署名更改为他人，根据《计算机软件保护条例》第二十三条第四款的规定，项目负责人的行为侵犯了张某的开发者身份权及署名权。

答案：A

11.2.3　同步练习

1．我国专利申请的原则之一是______。

A．申请在先　　　　B．申请在先与使用在先相结合

C．使用在先　　　　D．申请在先、使用在先或者二者相结合

2．李某在《电脑与编程杂志》上看到张某发表的一组程序，颇为欣赏，就复印了一百份作为程序设计辅导材料发给了学生。李某又将这组程序逐段加以评析，写成评论文章后寄到《电脑编程技巧》杂志上发表。李某的行为______。

A．侵犯了张某的著作权，因为其未经许可，擅自复印张某的程序

B．侵犯了张某的著作权，因为在评论文章中全文引用了发表的程序

C．不侵犯张某的著作权，其行为属于合理使用

D．侵犯了张某的著作权，因为其擅自复印，又在其发表的文章中全文引用了张某的程序

3．关于软件著作权产生的时间，表述正确的是______。

A．自软件首次公开发表时

B．自开发者有开发意图时

C．自软件得到国家著作权行政管理部门认可时

D．自软件完成创作之日起

4．李某大学毕业后在 M 公司销售部门工作，后由于该公司软件开发部门人手较紧，李某被暂调到该公司软件开发部开发新产品，两周后，李某开发出一种新软件。该软件著作权应归______所有。

A．李某　　B．M 公司　　C．李某和 M 公司　　D．软件开发部

5．若某人持有盗版软件，但他本人确实不知道该软件是盗版的，则______承担侵权责任。

A．应由该软件的持有者

B．应由该软件的提供者

C．应由该软件的提供者和持有者共同

D．该软件的提供者和持有者都不

6．______不属于知识产权的范围。

A．地理标志权　B．物权　C．邻接权　D．商业秘密权

7．　如果两名以上的申请人分别就同样的发明创造申请专利，专利权应授予＿＿＿＿。

A．最先发明的人　B．最先申请的人

C．所有申请人　D．协商后的申请人

11.2.4　同步练习参考答案

1．A　2．C　3．D　4．B

5．B　6．B　7．B

11.3　本 章 小 结

本章知识点在 2013 年的新大纲中改动不大，只是有一些描述方面的调整。另外在教材上，原来的第 11 章和第 12 章合并为一章，知识点没有那么精细了。

本章 11.1 节主要要求考生掌握标准和标准化知识，包括标准化的概念、标准化的过程模式、标准的分类、代号和编号、国际先进标准、信息技术标准化、标准化组织和 ISO 9000 标准简介。

11.1 节在近年上午试题中一般只出一道题，主要考查基本概念。在标准化知识中，考生要熟悉标准的分类、标准化的过程模式的区别、国际主要标准化组织及其主要制定的标准和进行的工作。对本节的内容只需多熟悉了解，一般不考查分析应用。

11.2 节主要要求考生掌握知识产权的概念与特点；计算机软件著作权的主体与客体；计算机软件受《著作权法》保护的条件；计算机软件著作权的权利，包括著作人身权、著作财产权、软件持有人的权利、著作权的行使和著作权的保护期，计算机软件著作权的归属，计算机软件著作权侵权的鉴别和侵权的法律责任，计算机软件的商业秘密权。专利权的有关概念。企业知识产权的保护等。在近年上午试题中一般也只出一道题。

11.2 节考过的知识点较多，而且出题以应用判断的类型较多，考生需要对具体的知识点理解，能灵活运用，作出正确判断。

11.4　达标训练题及参考答案

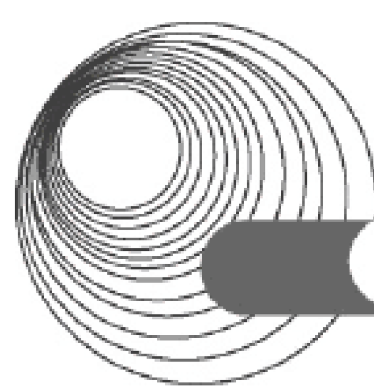
11.4.1　达标训练题

1．　甲、乙两人在同一时间就同样内容的发明创造提交了专利申请，专利局将分别向各申请人通报有关情况，并提出多种解决这一问题的办法，不能采用＿＿＿＿＿＿的办法。

A．两申请人作为同一件申请的共同申请人

B．其中一方放弃权利并从另一方得到适当的补偿

C．两件申请都不授予专利权



D．两件申请都授予专利权

2. 授予专利权的条件不包括________。

A．新颖性　　B．独一性　　C．创造性　　D．实用性

3. 利用单位物质条件但是在业余时间、在与本职工作无关的领域开发出的软件______。

A．属于开发者所在的单位

B．属于开发者个人

C．属于单位和个人共同所有

D．通过协商解决所有权

4. BS 7799 标准是由英国标准协会(BSI)制定的信息安全管理标准，是目前国际上具有代表性的信息安全管理体系标准，标准包括 BS7799-1:1999《信息安全管理实施细则》和 BS7799-2:1999_______两部分。

A．《信息安全服务体系规范》　　B．《信息安全管理体系规范》

C．《信息安全协议体系规范》　　D．《信息安全软件体系规范》

5. _______在通信领域最著名的研究成果是 802 局域网标准。802 标准定义了总线网络和环形网络的通信协议。

A．IEEE　　B．ISO　　C．IEC　　D．EIA

6．根据《国家标准管理办法》和《行业标准管理办法》，下列标准不属于强制性标准的是(1)。《计算机软件产品开发文件编制指南》(GB 8567—1988)属于(2)标准。编写标准的基本要求是：必须符合 GB/T1.1—2000(3)第 1 单元：标准的起草与表述规则。_____：《标准编写的基本规定》的规定。

(1) A．药品、食品卫生、兽药、农药和劳动卫生标准

B．产品生产、储运和使用中的安全及劳动安全标准

C．工程建设的质量、安全、卫生等标准

D．信息技术词汇分布式数据处理标准

(2) A．强制性国家　　B．推荐性国家　　C．强制性行业　　D．推荐性行业

(3) A．《中华人民共和国强制性国家标准》

B．《中华人民共和国推荐性国家标准》

C．《中华人民共和国标准化工作导则》

D．《中华人民共和国标准化指导性技术文件》

(4) A．第 1 部分　　B．第 2 部分　　C．第 3 部分　　D．第 4 部分

11.4.2 参考答案

1．D　　2．B　　3．D　　4．B

5．A　　6．(1) D　(2) A　(3) C　(4) A

第 12 章　计算机专业英语

大纲要求：

- 掌握计算机技术的基本词汇。
- 能正确阅读和理解计算机领域的英文资料。

12.1　专业英语试题分析

12.1.1　考点辅导

关于计算机英语，2006 年以后的试题中第 71～75 题一般是完形填空的形式，主要考查应试者结合计算机专业技术知识对全文综合理解的程度和串联上下文的能力；应试者语法知识和对句法结构的辨识能力；应试者的词汇量和词汇运用能力。

具体而言，完形填空主要考查应试者对语篇中句法、词语和短语的把握能力，具有较强的测试性。每一个空处都要通过上下文进行综合考虑，仅仅依靠一个单句往往无法确立正确选项。

语篇的内容往往是对网络技术中协议、通信过程、设备、最新技术等相关知识的描述，需要应试者对这些内容有一定的了解。

1. 完形填空中的句法

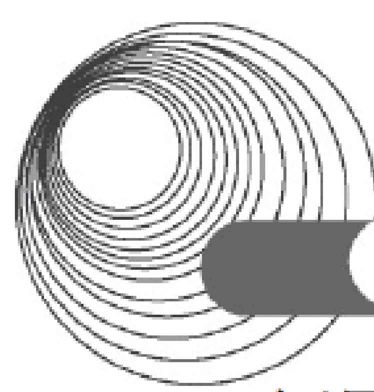
计算机英语的完形填空，句法强调时态、语态、倒装、复合，同时要求主语、谓语和宾语结构在数、格等方面的一致性。此外，连接手段包括关系代词、关系副词、连接词等，要求与整个语篇的行文相一致，起到或承接、或转折、或加强的作用，有着非常突出的个性特征。

时态在描述某项事务的发展历史时，一般采用过去时态；对目前尚在使用中的技术，采取完成时态或现在时；对未来技术的展望，大都采用将来时。句中几个受同一时间状态限制的动词时态在表达形式上要保持一致。这里包括并列的谓语动词以及主句和从句中谓语动词在表达形式上的一致。

计算机英语的语篇在描述技术类知识时，语态一般力求客观，采用描述性和被动语态较多。这里要注意只有及物动词及相当于及物动词的词组才有被动语态的表达形式。在并列结构中，同样的语义往往需要同样的语态表达形式。

2. 完形填空中的短语和固定用法

英语中有相当数量的动词短语、介词短语和固定搭配，其来源广泛，搭配方式丰富多变。因此需要应试者从动词入手，熟悉固定搭配，尤其是动词短语；从介词入手，了解介词本身的意义，进而了解同一个介词与不同动词、名词搭配产生的不同或相关的意义；理



解固定搭配的外延，增强对语义提示的审查力。

3. 完形填空的答题要领

关于完形填空要注意以下答题要领。

(1) 通过首句或出现的核心词汇来推断全文的信息。短文的首句往往是主题句，或出现了核心词汇，能为理解文章的大意和主要内容提供必要线索。一般首句还提供背景资料，因此要特别注意首句，抓住整个段落的纲要。

(2) 把握文章发展的基本线索。文章总是按照一定思路发展起来的，不同的逻辑关系主要依靠使用逻辑连接词来表达，文章如果没有出现内在的逻辑关系，就会出现语义不清、逻辑混乱。所以通过表示逻辑关系的词汇把握文章发展的基本线索是至关重要的。

借助语法知识和专业背景知识确定正确的词汇选项。

计算机专业英语词汇的考查在试题中占一定比例，词汇选项的设计和文章难度的制定与语法都息息相关。应试者务必借助语法知识和专业背景知识来确定正确的词汇选项。同时注意填入的词汇和文中句子的结构要求相一致。

4. 完形填空的答题步骤

关于完形填空题可采取以下答题步骤。

(1) 通读全文。完形填空是考查全面理解内容的基础上运用语言的能力，由于试题篇幅较短，完全有时间利用通读对全文内容有一个基本的了解。应试者要快速阅读段落，把握基本观点，通读时以浏览为主，可以忽略细节。

(2) 复读答题。在通读的基础上，应试者最好能立即复读，并结合选项，从语法结构、语义、词义、固定搭配等方面结合专业知识来考虑选项。选定之后，还需要回读。在整个答题过程中，切记全文的整体意义，保持思路的连贯性，从而作出正确选择。

(3) 重读检查。在确定所有选项以后，一定要重读全文，检查并核实每个选项在整篇文章中没有造成语义、结构、逻辑等方面的差错，确保短文是一个内容连贯、层次清晰、中心思想突出的整体。

12.1.2 典型例题分析

例 1 There is nothing in this world constant but inconstancy. --SWIFT

Project after project designs a set of algorithms and then plunges into construction of customer- deliverable software on a schedule that demands delivery of the first thing built.

In most projects, the first system built is (71) usable. It may be too slow, too big, awkward to use, or all three. There is no (72) but to start again, smarting but smarter, and build a redesigned version in which these problems are solved. The discard and (73) may be done in one lump, or it may be done piece-by-piece. But all large-system experience shows that it will be done. Where a new system concept or new technology is used, one has to build a system to throw away, for even the best planning is not so omniscient(全知的) as to get it right the first time.

The management question, therefore, is not whether to build a pilot system and throw it away. You will do that. The only question is whether to plan in advance to build a (74) , or to promise to

deliver the throwaway to customers. Seen this way, the answer is much clearer. Delivering that throwaway to customers buys time, but it does so only at the (75) of agony(极大痛苦) for the user, distraction for the builders while they do the redesign, and a bad reputation for the product that the best redesign will find hard to live down.

Hence plan to throw dne away; you will, anyhow. (2013 年 11 月试题 71~75)

(71) A．almost　B．often　C．usually　D．barely

(72) A．alternative　B．need　C．possibility　D．solution

(73) A．design　B．redesign　C．plan　D．build

(74) A．throwaway　B．system　C．software　D．product

(75) A．worth　B．value　C．cost　D．invaluable

解析：不变只是愿望，变化才是永恒。——SWIFT

一个接一个的软件项目都是一开始设计算法，然后将算法应用到待发布的软件中，接着根据时间进度把第一次开发的产品发布给客户。

对于大多数项目，第一个开发的系统并不适用。它可能太慢、太大、难以使用，或者三者兼有。要解决所有的问题，除了重新开始以外，没有其他的办法，即开发一个更灵巧或者更好的系统。系统的丢弃和重新设计可以一步完成，也可以一块块地实现。所有大型系统的经验都显示，这是必须完成的步骤。而且，新的系统概念或新技术会不断出现，因此开发的系统必须被抛弃，但即使是最优秀的项目计划也不能无所不知地在最开始就解决这些问题。

因此，管理上的问题不再是“是否构建一个实验性的系统，然后抛弃它”，你必须这样做。现在的问题是“是否预先计划抛弃原型的开发，或者是否将该原型发布给用户”。从这个角度看待问题，答案更加清晰。将原型发布给用户，虽然可以获得时间，但是其代价高昂——对于用户，使用极度痛苦；对于重新开发的人员，分散了精力；对于产品，影响了声誉，即使是最好的再设计也难以挽回名声。

因此，为舍弃而计划，无论如何，你一定要这样做。

答案：(71) D　(72) A　(73) B　(74) A　(75) C

例 2　Cloud computing is a phrase used to describe a variety of computing concepts that involve a large number of computers (71) through a real-time communication network such as the Internet. In science, cloud computing is a (72) for distributed computing over a network, and means the (73) to run a program or application on many connected computers at the same time.

The architecture of a cloud is developed at three layers: infrastructure, platform, and application.The infrastucture layer is built with virtualized compute storage and network resources. The platform layer is for general-purpose and repeated usage of the collection of software resources. The application layer is formed with a collection of all needed software modules for SaaS applications. The infrastucture layer serves as the (74) for building the platform layer of the cloud. In turn, the platform layer is foundation for implementing the (75) layer for SaaS application. (2014 年 5 月试题 71~75)

(71) A. connected　B. implemented　C. optimized　D. virtualized

(72) A. replacement　B. switch　C. substitute　D. synonym(同义词)

(73) A. ability　B. approach　C. function　D. method

(74) A. network　B. foundation　C. software　D. hardware

(75) A. resource　B. service　C. application　D. software

解析：云计算是一个用来描述各种计算概念的一个短语，计算概念涉及大量计算机通过实时通信的网络，如 Internet，连接在一起。在科学研究中，云计算是分布式网络计算的同义词，意味着有能力同时在多台互联的计算机上运行一个程序或应用。

云的结构分为 3 层，即基础设施、平台和应用。基础实施层由虚拟计算机存储和网络资源构成；平台层是具有通用性和复用性的软件资源的集合；应用层是云上针对 SaaS 应用的所有应用软件的集合。基础设施层是建立平台层的硬件基础；相应地，平台层是执行应用层 SaaS 应用的基础。

答案：(71) A　(72) D　(73) A　(74) D　(75) C。

例 3　At a basic level, cloud computing is simply a means of delivering IT resources as (71) . Almost all IT resources can be delivered as a cloud service: applications, compute power, storage capacity, networking, programming tools, even communication services and collaboration (72) .

Cloud computing began as large-scale Internet service providers such as Google, Amazon, and others built out their infrastructure. An architecture emerged: massively scaled, (73) distributed system resources, abstracted as virtual IT services and managed as continuously configured, pooled resources. In this architecture, the data is mostly resident on (74) "somewhere on the Internet" and the application runs on both the "cloud servers" and the user's browser.

Both clouds and grids are built to scale horizontally very efficiently. Both are built to withstand failures of (75) elements or nodes. Both are charged on a per-use basis. But while grids typically process batch jobs, with a defined start and end point, cloud services can be continuous. What's more, clouds expand the types of resources available-file storage, databases, and Web services-and extend the applicability to Web and enterprise applications.(2012 年上半年试题 71～75)

(71) A. hardware　B. computers　C. services　D. software

(72) A. computers　B. disks　C. machines　D. tools

(73) A. horizontally　B. vertically　C. inclined　D. decreasingly

(74) A. clients　B. middleware　C. servers　D. hard disks

(75) A. entire　B. individual　C. general　D. separate

解析：基本上，云计算仅仅意味着将 IT 资源作为服务来提供。几乎所有 IT 资源都可以作为一种云服务进行传递，如应用程序、计算能力、存储容量、网络、编程工具，设置是通信服务和协作工具。

云计算开始成为大规模的因特网服务提供商，如谷歌、亚马逊，以及其他一些建立了基础设施的公司。一种框架开始出现：不断扩大的、水平分布的系统资源，抽象成虚拟的 IT 服务，并不断进行配置和资源汇集的管理。在这种框架中，数据主要存储在服务器(因特网的某个地方)中，应用程序在云服务器和用户浏览器两端同时运行。

云技术和网格技术的建立都非常有效地扩大了规模，可以禁得住单个元素或者节点的失败。两者都受每次基点使用的控制。但是，网格主要处理批量的工作，有一个定义好的起点和终点，而云服务则是连续的。更为重要的是，云技术扩展了可获取资源(文件存储、

数据库、Web 服务)的类型，扩展了对 Web 和企业应用的适应性。

答案：(71) C　　(72) D　(73) A　(74) C　(75) B

例 4　Extreme Programming (XP) is a discipline of software development with __(71)__ of simplicity, communication, feedback and courage. Successful software development is a team effort - not just the development team, but the larger team consisting of customer, management and developers. XP is a simple process that brings these people together and helps them to successd together. XP is aimed primarily at object-oriented projects using teams of a dozen or fewer programmers in one location. The principles of XP apply to any __(72)__ project that needs to deliver quality software rapidly and flexibly.

An XP project needs a __(73)__ customer to provide guidance. Customers, programmers, managers, are all working __(74)__ to build the system that's needed. Customers - those who have software that needs to be developed - will learn simple, effective way to __(75)__ what they need, to be sure that they are getting what they need, and to steer the project to success. (2011 年下半年试题 71～75)

(71) A. importance　　B. keys　　C. role　　D. values
(72) A. small-sized　　B. moderately-sized
　　C. large-sized　　D. huge-sized
(73) A. part-time　　B. casual　　C. seldom　　D. full-time
(74) A. together　　B. by themselves　　C. separately　　D. alone
(75) A. tell　　B. know　　C. communicate　　D. feedback

解析：极限编程(XP)是一门软件开发学科，它基于简洁、沟通、反馈和勇气这样的价值观。成功的软件开发是一个团队努力的结果，这里的团队不仅仅是开发团队，而是包括顾客、管理者和开发者在内的更大的团队。XP 是一条可以使得一起开发软件的人们共同进步直至卓越的途径。XP 的目标主要是在面向对象的项目中利用若干团队或在某一特定领域的少数编程者。对任何中等规模的项目，XP 的法则是快速地、灵活地递交高品质的软件。

XP 项目需要一个全职的顾客来提供指导。顾客、编程者、管理员一起工作去建立所需要的系统。顾客(那些需要开发软件的人)应以简单、高效的方法去传达他们需要什么，以保证他们得到的正是自己所需要的，同时引导项目的成功。

答案：(71) D　　(72) B　(73) D　(74) A　(75) C

例 5　Teams are required for most engineering projects. Although some small hardware or software products can be developed by individuals, the scale and complexity of modem systems is such, and the demand for short schedules so great, that it is no longer __(71)__ for one person to do most engineering jobs. Systems development is a team __(72)__, and the effectiveness of the team largely determines the __(73)__ of the engineering.

Development teams often behave much like baseball or basketball teams. Even though they may have multiple specialties, all the members work toward __(74)__.However, on systems maintenance and enhancement teams, the engineers often work relatively independently, much like wrestling and track teams.

A team is (75) just a group of people who happen to work together. Teamwork takes practice and it involves special skills. Teams require common processes; they need agreed-upon goals; and they need effective guidance and leadership. The methods for guiding and leading such teams are well known, but they are not obvious.(2014 年 11 月试题 71~75)

(71) A. convenient B. existing C. practical D. real

(72) A. activity B. job C. process D. application

(73) A. size B. quality C. scale D. complexity

(74) A. multiple objectives B. different objectives C. a single objective D. independent objectives

(75) A. relatively B. / C. only D. more than

解析：大部分工程项目都需要团队来开发。尽管一些小的硬件或软件产品可以由个人开发，而大规模的、复杂的模型系统以及开发日程很紧迫的，由单个人去完成大部分的项目工作就不切实际了。系统开发是团队行为，团队的效率在很大程度上决定着项目的品质。开发团队的行为经常像棒球队或者篮球队。尽管每个人有很多特性，但所有成员朝着同一个目标工作。但是，在系统维护和加强团队方面，工程师的工作往往相对独立，更像摔跤或田径队。

团队不仅仅是一组一起工作的人。团队间需要协调，而协调需要特殊的技巧。团队需要共同的过程，需要协商好的目标，需要高效的指导和领导。指导和领导团队的方法是众所周知的，但并不明显。

答案：(71) C (72) A (73) B (74) C (75) D

例 6 People are indulging in an illusion whenever they find themselves explaining at a cocktail(鸡尾酒) party, say, that they are “in computers,”or“ in telecommunications,”or “in electronic funds transfer”. The implication is that they are part of the high-tech world. Just between us，they usually aren’t. The researchers who made fundamental breakthroughs in those areas are in a high-tech business. The rest of us are (71) of their work. We use computers and other new technology components to develop our products or to organize our affairs. Because we go about this work in teams and projects and other tightly knit working group(紧密联系在一起的工作小组), we are mostly in the human communication business. Our successes stem from good human interactions by all participants in the effort，and our failures stem from poor human interactions.

The main reason we tend to focus on the (72) rather than the human side of work is not because it’s more (73) , but because it’s easier to do. Getting the new disk drive installed is positively trivial compared to figurine out why Horace is in a blue funk(恐惧)or why Susan is dissatisfied with the company aver only a few months. Human interactions are complicated and never very crisp(干脆的，干净利落的) and clean in their effects, but they matter more than any other aspect of the work.

If you find yourself concentrating on the (74) rather than the (75) , you’re like the vaudeville character(杂耍人物)who loses his Keys on a dark street and looks for them on the adjacent street because, as he explains, “The light is better there!”(2010 年下半年试题 71～75)

(71) A．creators　B．innovators　C．appliers　D．inventors
(72) A．technical　B．classical　C．social　D．societal
(73) A．trivial　B．crucial　C．minor　D．insignificant
(74) A．technology　B．sociology　C．physiology　D．astronomy
(75) A．technology　B．sociology　C．physiology　D．astronomy

解析：如果人们发现自己在鸡尾酒会上侃侃而谈自己从事于计算机、无线电通信或者电子资金转账行业，此时他们正沉湎于幻想之中。他们的意思是自己是高科技世界的一份子。而这些人在我们之中，他们并不属于高科技行业的人。在这些领域中，只有取得根本性突破的研究者才能称得上是高科技行业中的人。剩下的我们只是他们的成果的应用者。我们使用计算机或者其他新的工艺元件去开发产品或者去处理事务。因为完成工作的是紧密联系在一起的工作小组，所以通常是处在人际交往行业中。我们的成功源于能够很好地处理与所有参与者之间的人际关系，失败则源于欠缺的人际交流。

工作中，我们倾向于关注技术而不是工作中人性的一面，这并不是因为技术更为重要，而是因为用起来简单。相对于指出为什么 Horace 不胜恐惧或者 Susan 最近几个月对公司不满的原因，安装一个新的硬盘驱动器是再简单不过的事了。人际交往是复杂的，它们的影响不会立竿见影，但是比工作中任何其他方面都要紧。

如果你发现自己专注于技术而不是社会学，你就像一个在黑暗的街上丢失了钥匙，而在邻街寻找的杂耍人员，按照他的解释，“那边的光线比较好。”

答案：(71) C　(72) A　(73) B　(74) A　(75) B

例 7　Observe that for the programmer, as for the chef, the urgency of the patron(顾客)may govern the scheduled completion of the task, but it cannot govern the actual completion. An omelette(煎鸡蛋), promised in two minutes, may appear to be progressing nicely. But when it has not set in two minutes, the customer has two choices—waits or eats it raw. Software customers have had (71) choices.

Now I do not think software (72) have less inherent courage and firmness than chefs, nor than other engineering managers. But false (73) to match the patron's desired date is much more common in our discipline than elsewhere in engineering. It is very (74) to make a vigorous, plausible, and job risking defense of an estimate that is derived by no quantitative method, supported by little data, and certified chiefly by the hunches of the managers.

Clearly two solutions are needed. We need to develop and publicize productivity figures, bug-incidence figures, estimating rules, and so on. The whole profession can only profit from (75) such data. Until estimating is on a sounder basis, individual managers will need to stiffen their backbones and defend their estimates with the assurance that their poor hunches are better than wish derived estimates.(2010 年上半年试题 71～75)

(71) A．no　B．the same　C．other　D．lots of
(72) A．Testers　B．constructors　C．managers　D．architects
(73) A．Tasks　B．jobs　C．Works　D．scheduling
(74) A．easy　B．difficult　C．simple　D．painless
(75) A．sharing　B．excluding　C．omitting　D．ignoring

解析：观察一下程序员，你可能会发现，如同厨师一样，某项任务的计划进度可能受限于顾客要求的紧迫程度，但这并不能控制实际的完成情况。就像约好在两分钟内完成一个煎蛋，可能看上去进行得很好。但是，当它无法在两分钟内完成时，顾客只能有两种选择：等待或者生吃它。软件顾客的情况与此类似。

现在我并不认为软件经理比厨师或者其他工程经理缺少内在的勇气和坚持。但是，为了满足顾客期望的日期而造成的不合理的进度安排，在软件领域比其他工程领域都要普遍得多。很难产生一个健壮的、可靠的和规避风险的估计，这是因为没有量化的方法，可以用来支持的数据很少，并要完全借助软件经理的直觉。

很显然，需要两种解决方案。我们需要开发和推行生产率图表、缺陷率图表、评估规则等。整个组织都会因为共享数据而获益。在基于可靠基础的估算出现之前，项目经理需要挺直腰杆并支持他们的估计，确信自己的经验和直觉总比从期望得出的估计要强得多。

答案：(71) B (72) C (73) D (74) B (75) A

例 8 Why is __(71)__ fun? What delights may its practitioner expect as his reward? First is the sheer joy of making things. As the child delights in his mud pie，so the adult enjoys building things，especially things of his own design. Second is the pleasure of making things that are useful to other people. Third is the fascination of fashioning complex puzzle-like objects of interlocking moving parts and watching them work in subtle cycles，playing out the consequences of principles built in from the beginning. Fourth is the joy of always learning，which springs from the __(72)__ nature of the task. In one way or another the problem is ever new，and its solver learns something:sometimes __(73)__，sometimes theoretical，and sometimes both. Finally，there is the delight of working in such a tractable medium. The __(74)__，like the poet，works only slightly removed from pure thought-stuff. Few media of creation are so flexible，so easy to polish and rework，so readily capable of realizing grand conceptual structures.

Yet the program __(75)__，unlike the poet's words， is real in the sense that it moves and works， producing visible outputs separate from the construct itself. It prints results，draws pictures， produces sounds， moves arms. Programming then is fun because it gratifies creative longings built deep within us and delights sensibilities we have in common with all men.(2009 年下半年试题 71～75)

(71) A．programming B．composing C．working D．writing
(72) A．repeating B．basic C．non-repeating D．advance
(73) A．semantic B．practical C．lexical D．syntactical
(74) A．poet B．architect C．doctor D．programmer
(75) A．construct B．code C．size D．scale

解析：为什么编程是如此有趣的事？是什么乐趣促使编程者从事该行业？首先是纯粹的做事情的乐趣。正如小孩子对泥饼感兴趣一样，成人也热衷于构建事物，尤其是自己设计的事物。第二是自己做出来的东西对别人有帮助的那种兴奋感。第三是着迷于塑造复杂的如谜题一般的连锁移动部件，并且观察它们微妙地循环工作着，从先前定义好的规则中演绎出结果。第四是不断学习的乐趣，它来自任务的不重复性。从一种或是另一种角度，这个问题曾经是新的，但它的解决者学到了一些东西：一些实用的东西，一些理论上的东

西，或者两者兼具的东西。最后是在一种易用的媒介上工作的乐趣。编程者，像诗人一样，从纯粹的思想中提炼出成品。很少有媒体的创作是如此灵活，如此容易加以润饰和返工，有如此轻易实现宏伟概念结构的能力。

然而编程的构建，并不像诗人的词句，在某种意义上来说，它是真实移动和工作的，从构建的本身过程中产生出可视化的输出。它打印输出，画图，产生声音，移动。由此可见，编程是有趣的，因为它使长期深埋在我们中的创造性得到发挥，使我们共同具有的感情得到宣泄。

答案：(71) A　(72) C　(73) B　(74) D　(75) A

例 9　Why Have Formal Documents?

Finally, writing the decisions down is essential. Only when one writes do the gaps appear and the __(71)__ protrude(突出). The act of writing turns out to require hundreds of mini-decisions, and it is the existence of these that distinguishes clear, exact policies from fuzzy ones.

Second, the documents will communicate the decisions to others. The manager will be continually amazed that policies he took for common knowledge are totally unknown by some member of his team. Since his fundamental job is to keep everybody going in the __(72)__ direction, his chief daily task will be communication, not decision-making, and his documents will immensely __(73)__ this load.

Finally, a manager's documents give him a data base and checklist. By reviewing them __(74)__ he sees where he is, and he sees what changes of emphasis or shifts in direction are needed.

The task of the manager is to develop a plan and then to realize it. But only the written plan is precise and communicable. Such a plan consists of documents on what, when, how much, where, and who. This small set of critical documents __(75)__ much of the managers work. If their comprehensive and critical nature is recognized in the beginning, the manager can approach them as friendly tools rather than annoying busywork. He will set his direction much more crisply and quickly by doing so.(2015 年 5 月试题 71~75)

(71)　A. inconsistencies　B. consistencies　C. steadiness　D. adaptability
(72)　A. other　B. different　C. another　D. same
(73)　A. extend　B. broaden　C. lighten　D. release
(74)　A. periodically　B. occasionally　C. infrequently　D. rarely
(75)　A. decides　B. encapsulates　C. realizes　D. recognizes

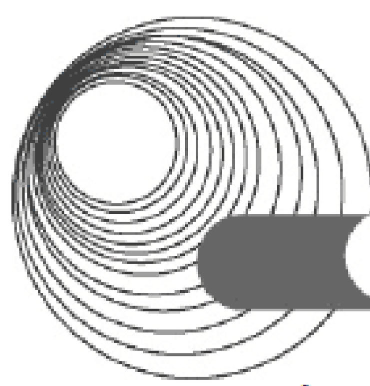
解析：为什么要有正式的文档？

首先，将决策写下来是关键的。只有写出后差距才能出现，矛盾才会突出。写的过程需求成百上千的小决策，这些小决策的存在将清楚的、准确的决策从模糊中区分出来。

其次，文档将会用于与其他人进行决策交流。管理者将会不断感到惊奇的是，他所采取的常识性政策，有些团队成员竟全然不知。既然他的基本工作是使每个人朝同一个方向前进，他的主要工作就是交流，而不是制定决策，他的文档能很好地减轻这个负担。

最后，管理者的文档给他提供了一个数据库和检验表。通过定期回顾他能知道自己所处的位置，并知道需要重点改变什么或方向如何变化。

管理者的任务是制定一个计划，然后去实现它。但只有书面计划才是清晰的、可交流



的。这样的一个计划应该由包括“什么”“什么时候”“多少”“在哪里”“谁”这样的文档组成。这个小的关键文档的集合决定了管理者的工作量。如果它们的综合性和关键性在一开始就被认识到，管理者将会把它们作为友好的工具，而不是烦躁的工作。这样做，他就会更加清晰地确定方向。

答案：(71) A (72) D (73) C (74) A (75) A

例 10 Should go without saying that the focus of UML is modeling. However, what that means, exactly, can be an open-ended question. (71) is a means to capture ideas, relationships, decisions, and requirements in a well-defined notation that can be applied to many different domains. Modeling not only means different things to different people, but also it can use different pieces of UML depending on what you are trying to convey. In general, a UML model is made up of one or more (72). A diagram graphically represents things, and the relationships between these things. These (73) can be representations of real-world objects, pure software constructs, or a description of the behavior of some other objects. It is common for an individual thing to show up on multiple diagrams; each diagram represents a particular interest, or view, of the thing being modeled. UML 2.0 divides diagrams into two categories: structural diagrams and behavioral diagrams. (74) are used to capture the physical organization of the things in your system, i.e., how one object relates to another. (75) focus on the behavior of elements in a system. For example, you can use behavioral diagrams to capture requirements, operations, and internal state changes for elements. (2008 年下半年试题 71～75)

(71) A．Programming B．Analyzing C．Designing D．Modeling

(72) A．views B．diagrams C．user views D．structure pictures

(73) A．things B．pictures C．languages D．diagrams

(74) A．Activity diagrams B．Use-case diagrams
C．Structural diagrams D．Behavioral diagrams

(75) A．Activity diagrams B．Use-case diagrams
C．Structural diagrams D．Behavioral

解析：不用说，UML 的聚焦点是建模。然而，这就意味着它是个开发式问题。建模是一种以规范好的符号(这些符号可以应用于许多不同的领域)记录概念、关系、决策和需求的方法。建模不仅意味着对不同的人有不同的事物，也可以根据你的需求来使用不同的 UML 块。一般意义上，UML 由一个或多个图组成。图能够形象化地表示事物以及这些事物之间的联系。这些事物能代表现实世界中的物体、纯软件结构或者一些其他物体行为的描述。在 UML 中用多个图形表示单个物体是非常普遍的；每个图形代表被建模事物的某个特定的兴趣、观点。UML 2.0 将图形划分为两个范畴，即结构图和行为图。结构图用于记录系统中事物的物理组织以及与其他模块怎样联系。行为图则把重心放在系统中元素的行为上，如可以用行为图记录需求、操作以及元素的内部状态的变化。

答案：(71) D (72) B (73) A (74) C (75) D

例 11 Object-oriented analysis (OOA) is a semiformal specification technique for the object-oriented paradigm. Object-oriented analysis consists of three steps. The first step is (71). It determines how the various results are computed by the product and presents this information in

the form of a __(72)__ and associated scenarios. The second is __(73)__, which determines the classes and their attributes, then determines the interrelationships and interaction among the classes. The last step is __(74)__, which determines the actions performed by or to each class or subclass and presents this information in the form of __(75)__. (2008 年上半年试题 71～75)

(71) A. use-case modeling　　B. class modeling
C. dynamic modeling　　D. behavioral modeling

(72) A. collaboration diagram　　B. sequence diagram
C. use-case diagram　　D. activity diagram

(73) A. use-case modeling　　B. class modeling
C. dynamic modeling　　D. behavioral modeling

(74) A. use-case modeling　　B. class modeling
C. dynamic modeling　　D. behavioral modeling

(75) A. activity diagram　　B. component diagram
C. sequence diagram　　D. state diagram

解析：面向对象分析是一个面向对象的半正式化的规范技术。它由 3 个阶段组成。第一阶段是用例建模。通过用例建模能计算出多种结果，并以用例图和相关场景的形式体现这些信息。第二阶段是类建模，通过类建模能够确定类及其属性，在此基础上就能确定类之间的内部联系和交互作用。最后一阶段是动态建模，通过动态建模能够确定每个类及其子类的行为作用，并以状态图来体现这些信息。

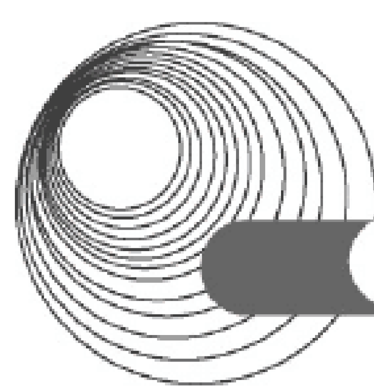
答案：(71) B　(72) C　(73) D　(74) C　(75) C

例 12　In a world where it seems we already have too much to do, and too many things to think about, it seems the last thing we need is something new that we have to learn.

But use cases do solve a problem with requirements: with __(71)__ declarative requirements it's hard to describe steps and sequences of events.

Use cases, stated simply, allow description of sequences of events that, taken together, lead to a system doing something useful. As simple as this sounds, this is important. When confronted only with a pile of requirements, it's often __(72)__ to make sense of what the authors of the requirements really wanted the system to do. In the preceding example, use cases reduce the ambiguity of the requirements by specifying exactly when and under what conditions certain behavior occurs; as such, the sequence of the behaviors can be regarded as a requirement. Use cases are particularly well suited to capture approaches. Although this may sound simple, the fact is that __(73)__ requirement capture approaches, with their emphasis on declarative requirements and "shall" statements, completely fail to capture the __(74)__ of the system's behavior. Use cases are a simple yet powerful way to express the behavior of the system in way that all stakeholders can easily understand.

But, like anything, use cases come with their own problems, and as useful as they are, they can be __(75)__. The result is something that is as bad, if not worse, that the original problem. Then it's important to utilize use cases effectively without creating a greater problem than the one you started with.(2015 年 11 月试题 71~75)



(71) A. plenty　B. loose　C. extra　D. strict
(72) A. impossible　B. possible　C. sensible　D. practical
(73) A. modern　B. conventional　C. different　D. formal
(74) A. statics　B. nature　C. dynamics　D. originals
(75) A. misapplied　B. applied　C. used　D. powerful

解析：在这个世界上，似乎我们已经有太多的事情要做，有太多的事情要思考，最后需要的是要我们去学习的新事物。

但是，用例确实解决了需求问题：有大量声明性的需求，难以去描述事件的步骤和序列。

简单地说，用例允许描述事件的序列，一起考虑，引导系统做一些有用的事情。这听起来很简单，但很重要。当面对一对需求时，通常不可能弄清楚用户想要系统做什么。在前面的例子中，用例通过准确描述在什么时候什么情况下某些行为发生，减少了需求的模糊性；这样，行为序列可以被视为需求。用例特别适合捕捉方法。虽然这听起来简单，事实是，由于不同的需求捕获方法强调不同的声明性需求和“应该有”的状态，因此完全无法捕捉系统的静态行为。用例是一种简单而有力的描述系统行为的方式，这种方式很容易被所有的利益相关者理解。

但是，和所有事物一样，用例很有用，功能强大，但也有自身的问题。结果是，一些事情会和原来的问题一样糟，如果没有变得更糟。因此，重要的是高效地使用用例，而不引入比开始时更大的问题。

答案：(71) A　(72) A　(73) C　(74) A　(75) D

例 13　(71) analysis emphasizes the drawing of pictorial system models to document and validate both existing and/or proposed systems. Ultimately, the system models become the (72) for designing and constructing an improved system. (73) is such a technique. The emphasis in this technique is process-centered. Systems analysts draw a series of process models called (74). (75) is another such technique that integrates data and process concerns into constructs called objects. (2007 年上半年试题 71～75)

(71) A. Prototyping　B. Accelerated　C. Model-driven　D. Iterative
(72) A. image　B. picture　C. layout　D. blueprint
(73) A. Structured analysis　B. Information Engineering
　C. Discovery Prototyping　D. Object-Oriented analysis
(74) A. PERT　B. DFD　C. ERD　D. UML
(75) A. Structured analysis　B. Information Engineering
　C. Discovery Prototyping　D. Object-Oriented analysis

解析：原型分析强调图案系统模式的牵引作用，以记录和验证现有和/或拟议的系统。最终，该系统模型为设计和构造一个改进的系统而成为图像。结构化分析是强调过程为中心的一种技术。系统分析绘制了一系列的过程模型，称为 DFD(数据流程图)。面向对象的分析也是这样一种技术，对象集成了数据和过程。

答案：(71) C　(72) D　(73) A　(74) B　(75) D

例 14　NAC's(Network Access Control) role is to restrict network access to only compliant

endpoints and __(66)__ users. However，NAC is not a complete LAN __(67)__ solution; additional proactive and __(68)__ security measures must be implemented. Nevis is the first and only comprehensive LAN security solution that combines deep security processing of every packet at 10Gbps，ensuring a high level of security plus application availability and performance. Nevis integrates NAC as the first line of LAN security __(69)__ In addition to NAC，enterprises need to implement role-based network access control as well as critical proactive security measures—real-time，multilevel __(70)__ inspection and microsecond threat containment. (2006 年下半年试题 66～70)

(66) A．automated B．distinguished C．authenticated D．destructed
(67) A．crisis B．security C．favorable D．excellent
(68) A．constructive B．reductive C．reactive D．productive
(69) A．defense B．intrusion C．inbreak D．protection
(70) A．port B．connection C．threat D．insurance

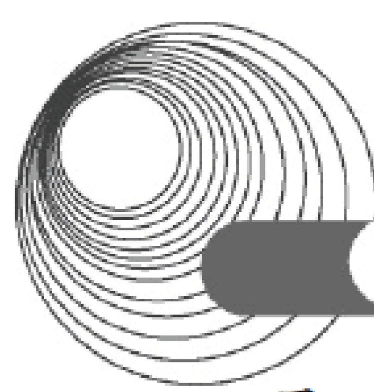
解析：NAC(网络访问控制)的作用是限制网络访问，只有那些被允许的终端和被授权的用户才可以。然而，NAC 不是一个完善的保证局域网安全的解决方案；预防性的和反应性的额外安全措施仍需完善。Nevis 是最早的和唯一全面的局域网安全解决方案，它深层次地考虑到了将来以 10Gb/s 的速率传输数据包时的安全问题，在确保好的应用性和性能的同时确保了高水平的安全性。Nevis 集成了 NAC，使它成为局域网安全防御的第一道防线。除了 NAC 之外，企业既需要完善基于角色的网络访问控制，也要完善关键的预防性安全措施——实时的、多层次的威胁检测和微秒级的威胁遏制。

答案：(66) C (67) B (68) C (69) A (70) C

例 15 Virtualization is an approach to IT that pools and shares __(71)__ so that utilization is optimized and supplies automatically meet demand. Traditional IT environments are often silos，where both technology and human __(72)__ are aligned around an application or business function. With a virtualized __(73)__，people，processes，and technology are focused on meeting service levels，__(74)__ is allocated dynamically，resources are optimized，and the entire infrastructure is simplified and flexible. We offer a broad spectrum of virtualization __(75)__ that allows customers to choose the most appropriate path and optimization focus for their IT infrastructure resources. (2006 年下半年试题 71～75)

(71) A．advantages B．resources C．benefits D．precedents
(72) A．profits B．costs C．resources D．powers
(73) A．system B．infrastructure C．hardware D．link
(74) A．content B．position C．power D．capacity
(75) A．solutions B．networks C．interfaces D．connections

解析：虚拟化是一种 IT 技术，它可以集成和共享资源，从而使得资源利用得到优化并能自动地满足需求。传统的 IT 环境中，技术和人力资源仅仅围绕着一种应用或者一种商用功能，并不能很好地联系起来。利用虚拟化的 IT 基础设施，人、方法和技术集中在一起满足不同的服务需求，负载能力被很好地动态分配，资源利用被优化，并且整个基础设施更加简化和灵活。我们提供了一个广泛的虚拟化解决方案，它允许客户选择最适合的途径，



最优化他们的 IT 基础设施资源的方案。

答案：(71) B　(72) C　(73) B　(74) D　(75) A

例 16　Originally introduced by Netscape Communications. (66) are a general mechanism which HTTP Server side applications，such as CGI (67)，can use to both store and retrieve information on the HTTP (68) side of the connection. Basically，Cookies can be used to compensate for the (69) nature of HTTP. The addition of a simple，persistent，client-side state significantly extends the capabilities of WWW-based (70). (2006 年上半年试题 66～70)

(66) A．Browsers　B．Cookies　C．Connections　D．Scripts
(67) A．graphics　B．processes　C．scripts　D．texts
(68) A．Client　B．Editor　C．Creator　D．Server
(69) A．fixed　B．flexible　C．stable　D．stateless
(70) A．programs　B．applications　C．rameworks　D．constrains

解析：Cookies 最初被网景通信引进，是 HTTP 服务器端应用程序的一种普遍通信机制。例如，CGI(公共网关接口)脚本，它能够用来在 HTTP 连接的客户端上存储和检索信息。在根本上，Cookies 能够用来弥补 HTTP 不保存客户端状态的缺陷。它通过简单、持续地维护客户端状态，明显地扩展了基于 WWW 的应用程序的能力。

答案：(66)　B　(67)　C　(68)　A　(69)　D　(70)　B

例 17　WebSQL is a SQL-like (71) language for extracting information from the web. Its capabilities for performing navigation of web (72) make it a useful tool for automating several web-related tasks that require the systematic processing of either all the links in a (73) ,all the pages that can be reached from a given URL through (74) that match a pattern，or a combination of both. WebSQL also provides transparent access to index servers that can be queried via the Common (75) Interface. (2006 年上半年试题 71～75)

(71) A．query　B．transaction　C．communication　D．programming
(72) A．browsers　B．servers　C．hypertexts　D．clients
(73) A．hypertext　B．page　C．protocol　D．operation
(74) A．paths　B．chips　C．tools　D．directories
(75) A．Router　B．Device　C．Computer　D．Gateway

解析：WebSQL 是一种类似于 SQL 的查询语言，它用于提取来自 Web 的信息。能够为 Web 超文本导航的这种能力使得它成为一个有用的工具，用于自动操作一个页面中有关链接，搜索通过匹配某种模式的路径从所给的 URL 中可以到达的所有的网页，或者是两者的结合。WebSQL 也提供对索引服务器的透明性访问，这种服务器可以通过公共网关接口来进行询问。

答案：(71) A　(72) C　(73) B　(74) A　(75) D

12.1.3　同步练习

1．Should go without saying that the focus of UML is modeling. However, what that means, exactly, can be an open-ended question. (1) is a means to capture ideas, relationships, decisions,

and requirements in a well-defined notation that can be applied to many different domains. Modeling not only means different things to different people, but also it can use different pieces of UML depending on what you are trying to convey. In general, a UML model is made up of one or more (2) . A diagram graphically represents things, and the relationships between these things. These (3) can be representations of real-world objects, pure software constructs, or a description of the behavior of some other objects. It is common for an individual thing to show up on multiple diagrams; each diagram represents a particular interest, or view, of the thing being modeled. UML 2.0 divides diagrams into two categories: structural diagrams and behavioral diagrams. (4) are used to capture the physical organization of the things in your system, i.e., how one object relates to another. (5) focus on the behavior of elements in a system. For example, you can use behavioral diagrams to capture requirements, operations, and internal state changes for elements.

(1) A. Programming　B. Analyzing　C. Designing　D. Modeling
(2) A. views　B. diagrams　C. user views　D. structure pictures
(3) A. things　B. pictures　C. languages　D. diagrams
(4) A. Activity diagrams　B. Use-case diagrams
C. Structural diagrams　D. Behavioral diagrams
(5) A. Activity diagrams　B. Use-case diagrams
C. Structural diagrams　D. Behavioral

2．Object-oriented analysis (OOA) is a semiformal specification technique for the object-oriented paradigm. Object-oriented analysis consists of three steps. The first step is (1) . It determines how the various results are computed by the product and presents this information in the form of a (2) and associated scenarios. The second is (3) ，which determines the classes and their attributes，then determines the interrelationships and interaction among the classes. The last step is (4) ，which determines the actions performed by or to each class or subclass and presents this information in the form of (5) .

(1) A. use-case modeling　B. class modeling
C. dynamic modeling　D. behavioral modeling
(2) A. collaboration diagram　B. sequence diagram
C. use-case diagram　D. activity diagram
(3) A. use-case modeling　B. class modeling
C. dynamic modeling　D. behavioral modeling
(4) A. use-case modeling　B. class modeling
C. dynamic modeling　D. behavioral modeling
(5) A. activity diagram　B. component diagram
C. sequence diagram　D. state diagram

3．The Rational Unified Process (RUP) is a software engineering process,which captures many of best practices in modern software development. The notions of (1) and scenarios have been proven to be an excellent way to capture function requirements. RUP can be described in two dimensions – time and content. In the time dimension，the software lifecycle is broken into cycles. Each cycle is divided into four consecutive (2) which is concluded with a well-defined (3) and can

be further broken down into (4) – a complete development loop resulting in a release of an executable product,a subset of the final product under development, which grows incrementally to become the final system. The content structure refers to the disciplines，which group (5) logically by nature.

(1) A. artifacts B. use-cases C. actors D. workers
(2) A. orientations B. views C. aspects D. phases
(3) A. milestone B. end-mark C. measure D. criteria
(4) A. rounds B. loops C. iterations D. circularities
(5) A. functions B. workflows C. actions D. activities

4. (1) analysis emphasizes the drawing of pictorial system models to document and validate both existing and/or proposed systems. Ultimately，the system models become the (2) for designing and constructing an improved system. (3) is such a technique. The emphasis in this technique is process-centered. Systems analysts draw a series of process models called (4) . (5) is another such technique that integrates data and process concerns into constructs called objects.

(1) A. Prototyping B. Accelerated C. Model-driven D. Iterative
(2) A. image B. picture C. layout D. blueprint
(3) A. Structured analysis B. Information Engineering
C. Discovery Prototyping D. Object-Oriented analysis
(4) A. PERT B. DFD C. ERD D. UML
(5) A. Structured analysis B. Information Engineering
C. Discovery Prototyping D. Object-Oriented analysis

5. NAC's(Network Access Control) role is to restrict network access to only compliant endpoints and (1) users. However,NAC is not a complete LAN (2) solution; additional proactive and (3) security measures must be implemented. Nevis is the first and only comprehensive LAN security solution that combines deep security processing of every packet at 10Gbps，ensuring a high level of security plus application availability and performance. Nevis integrates NAC as the first line of LAN security (4) In addition to NAC，enterprises need to implement role-based network access control as well as critical proactive security measures— real-time，multilevel (5)inspection and microsecond threat containment.

(1) A. automated B. distinguished C. authenticated D. destructed
(2) A. crisis B. security C. favorable D. excellent
(3) A. constructive B. reductive C. reactive D. productive
(4) A. defense B. intrusion C. inbreak D. protection
(5) A. port B. connection C. threat D. insurance

6. Virtualization is an approach to IT that pools and shares (1) so that utilization is optimized and supplies automatically meet demand. Traditional IT environments are often silos, where both technology and human (2) are aligned around an application or business function. With a virtualized (3) ,people,processes，and technology are focused on meeting service levels, (4) is allocated dynamically,resources are optimized，and the entire infrastructure is simplified and

flexible. We offer a broad spectrum of virtualization (5) that allows customers to choose the most appropriate path and optimization focus for their IT infrastructure resources.

(1)	A. advantages	B. resources	C. benefits	D. precedents
(2)	A. profits	B. costs	C. resources	D. powers
(3)	A. system	B. infrastructure	C. hardware	D. link
(4)	A. content	B. position	C. power	D. capacity
(5)	A. solutions	B. networks	C. interfaces	D. connections

7．Originally introduced by Netscape Communications． (1) are a general mechanism which HTTP Server side applications，such as CGI (2) ,can use to both store and retrieve information on the HTTP (3) side of the connection. Basically,Cookies can be used to compensate for the (4) nature of HTTP. The addition of a simple，persistent，client-side state significantly extends the capabilities of WWW-based (5) .

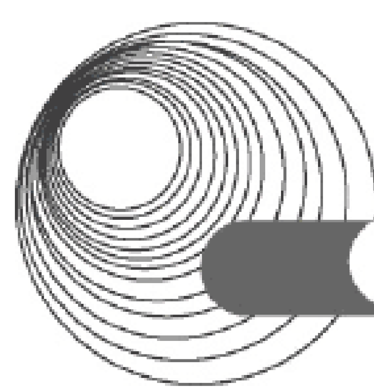
(1)	A. Browsers	B. Cookies	C. Connections	D. Scripts
(2)	A. graphics	B. processes	C. scripts	D. texts
(3)	A. Client	B. Editor	C. Creator	D. Server
(4)	A. fixed	B. flexible	C. stable	D. stateless
(5)	A. programs	B. applications	C. rameworks	D. constrains

8．WebSQL is a SQL-like (1) language for extracting information from the web. Its capabilities for performing navigation of web (2) make it a useful tool for automating several web-related tasks that require the systematic processing of either all the links in a (3) ,all the pages that can be reached from a given URL through (4) that match a pattern，or a combination of both. WebSQL also provides transparent access to index servers that can be queried via the Common (5) Interface.

(1)	A. query	B. transaction	C. communication	D. programming
(2)	A. browsers	B. servers	C. hypertexts	D. clients
(3)	A. hypertext	B. page	C. protocol	D. operation
(4)	A. paths	B. chips	C. tools	D. directories
(5)	A. Router	B. Device	C. Computer	D. Gateway

12.1.4 同步练习参考答案

1.	(1) D	(2) B	(3) A	(4) C	(5) D
2.	(1) B	(2) C	(3) D	(4) C	(5) C
3.	(1) B	(2) D	(3) A	(4) C	(5) D
4.	(1) C	(2) D	(3) A	(4) B	(5) D
5.	(1) C	(2) B	(3) C	(4) A	(5) C
6.	(1) B	(2) C	(3) B	(4) D	(5) A
7.	(1) B	(2) C	(3) A	(4) D	(5) B
8.	(1) A	(2) C	(3) B	(4) A	(5) D



12.2 本 章 小 结

本章知识点在 2013 年的新大纲中改动不大，只是有一些描述方面的调整。

本章考纲主要要求使用计算机英语看懂中等难度的计算机科学技术文章，形式是选择填空和完形填空。选择填空往往是在单句中给出一些技术中基本技术的定义和特征，要求选择对应的专用名词或缩写。完形填空一般是给出对一种技术或应用的原理和方法的具体描述，要求根据上下文选择完成短文。

关于本章内容的考试比例，从 2009 年开始，每年考 5 个小题，每套上午试题的第 71～75 题是对专业英语的考查。

无论是选择填空还是完形填空，都要求考生不仅掌握相当广泛的计算机技术知识，同时也要求有一定的专业英语水平。试题将两点结合起来，可以说是用英语的语言考查一些基础难度计算机科学技术知识试题。

12.3 达标训练题及参考答案

12.3.1 达标训练题

1. In the following essay,each blank has four choices. Choose the best answer and writedown on the answer sheet.

Microwave communication uses high-frequency(1)waves that travel in straightlines through the air. Because the waves cannot (2) with the curvature of the earth，they can be (3) only over short distance. Thus，microwave is a good (4) for sending data between buildings in a city or on a large college campus. For longer distances，the waves must be relayed by means of “dishes” or (5). These can be installed on towers，highbuildings,and mountain tops.

(1) A. optical　B. radio　C. electrical　D. magnetic

(2) A. reflect　B. distort　C. bend　D. absorb

(3) A. transmitted　B. transformed　C. converted　D. delivered

(4) A. material　B. equipment　C. medium　D. channel

(5) A. repeaters　B. radars　C. telephones　D. antennas

2. For each blank,choose the best answer from the four choices and write down on the answer sheet.

(1)is a protocol that a host uses to inform a router when it join or leaves an Internet multicat group.

(2) is an error detection code that most data communication networks use.

(3) is an interior gateway protocol that uses a distance vector algorithm to propagate routing information.

(4) is a transfer mode in which all types of information are organized into fixed form cells on an asynchronous or non-periodic basis over a range media .

(5) is an identifier of a web page.

(1) A．ICMP B．SMTP C．IGMP D．ARP

(2) A．4B/5B B．CRC C．Manchester D．Huffman Code

(3) A．OSPF B．RIP C．RARP D．BGP

(4) A．ISDN B．X.25 C．Frame Relay D．ATM

(5) A．HTTP B．URL C．HTML D．TAG

3．In the flowing essay,each blank has four choices. Choose the best answer and write down on the answer sheet.

Spread spectrum simply means that data is sent in small pieces over a number of the (1) frequencies available for use at any time in the specified rang. Devices using (2) spread spectrum(DSSS) communicate by (3) each byte of data into several parts and sending them concurrently on different (4) . DSSS uses a lot of the available (5) ,about 22 megaherts (MHz).

(1) A．continuous B．high C．low D．discrete

(2) A．direct-sequence B．discrect-sequence

C．duplicate-sequence D．dedicate-sequence

(3) A．splitting B．combining C．packing D．compacting

(4) A．bits B．frequencies C．packets D．messages

(5) A．rate B．velocity C．bandwidth D．period

12.3.2 参考答案

1．(1) B (2) C (3) A (4) C (5) D

解析：此题涉及微波传输方式和性质。

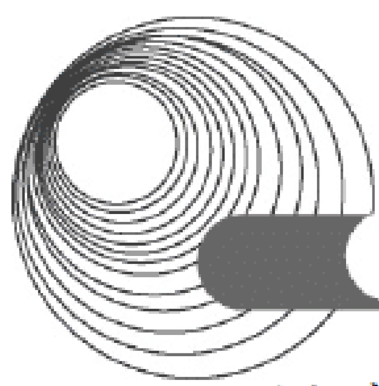
(1) 此题需要了解：microwave 意为微波，是无线电波的一种；radio wave 意为无线电波；optical wave 意为光波；electrical wave 意为电脉冲信号；magnetic wave 意为磁场波。

(2) 此句说明了无线电波传输的弱点，即不能随着地球的曲度 curvature 而弯曲变化，所以只能传输较短距离。reflect 意为反射；distort 意为扭曲、歪曲；bend 意为弯曲、使朝向，有随着弧度变化的意识；absorb 意为吸收。根据上下文结构，这里说的是一般无线电波传输的局限，从而突出下文微波传输的优势。

(3) 此题对 4 个意义相近的词语进行辨析考查。transmit 意为发送、传输，联系上下文符合发送电波题义；transform 意为转变，说明主语在性质上发生了变化；convert 意为使改变以适应；deliver 意为发送，但一般对象是物理上的有质量的物体。

(4) 此题也是对 4 个意义相近的词语进行辨析考查。根据上文，判定选项是说明微波性质的。material 意为材料；equipment 意为设备；medium 意为传输的媒介，符合题意；channel 意为频道。

(5) 根据上文中关键词 relay 意为接力、中继，by means of 意为可用方法，可以判定选项是用于转发或加强微波的设备，与 repeaters、radars、telephones 相比，天线 antenna 作为



微波的发射装置更为合适，而且下文也给出了天线安装位置的描述。

2．(1) C　(2) B　(3) B　(4) D　(5) B

解析：此题是对专业词语定义解释。涉及了路由协议知识、纠错验证码知识、网络传输模式等知识。知识点分散，要求全面掌握网络技术各方面的知识。

(1) 根据下文 Internet multicat group 可以透露出选项 IGMP 中的 G 为 group。IGMP (Internet Group Management Protocol，互联网组管理协议)使路由器通知其组播组的成员。ICMP 是 Internet 控制协议；SMTP 是邮件传输协议；ARP 是底层地址解析协议。

(2) CRC(Cyclic Redundancy Check，循环冗余检验)是用来验证数据帧中数据准确性的算法。an error detection code 给出了其性质的描述。

(3) Routing Information Protocol(路由信息协议，RIP)具有 interior gateway protocol 内部路由协议和 a distance vector algorithm 距离向量算法的性质。

(4) 此题考查各种网络传输模式。ISDN(Integrated Services Digital Network，综合业务数字网)是一种由 ITU 建议的，用来通过数字线路传输数据的国际标准，ISDN 使用电话载波线路和拨号连接，使用数字线路和交换设备。X.25 是一个模拟包交换的广域网技术，适合远距离传输数据，ITU 在 20 世纪 70 年代中期对其进行了标准化，X.25 可以支持 56Kb 带宽，它最早被用来在主机和远程终端之间通信。frame relay(帧中继)是一种升级的、数字版本的 X.25，它是基于数据包交换的，因为帧中继是数字传输，它最大可支持 1.544Mb/s 的传输速率，大于 X.25 的带宽，它提供许多 Internet 连接的基础。ATM(Asynchronous Transfer Mode，异步传输模式)，它靠定长的数据包使传输速率达到 25～622Mb/s，定长的数据包由 48B 的数据加 5B 的头信息组成，定长的数据包允许 ATM 在宽带应用上提供可预测流量的模型和更好的控制。asynchronous(异步)是此题的关键词。

(5) URL(Uniform Resource Locator，统一资源定位符)是一种标准地用来标识每个 Web 页面的方法，标识服务、主机名和 HTML 页或脚本名。其他选项都不具备 Web identifier 的性质。

3．(1) D　(2) A　(3) A　(4) B　(5) C

解析：此题涉及 spread spectrum(扩频)知识。它是一种类型的射频传输，使用低电平的、扩展在多个频率的信号。扩频比一般的窄带传输安全性高。了解这一原理对于全文的理解和选项的判断至关重要。

(1) 扩频使用低电平的、扩展在多个频率的信号，故选 discrete frequencies。discrete 意为离散的、分离的；continuous 意为连续的。

(2) 此题考查 4 个相近的词语辨析。direct 意为直接；discrete 意为离散的、分离的；duplicate 意为双重的、相同的；dedicate 意为致力、奉献。

(3) 此句说明了利用扩频发送数据的方式。by 后跟动名词意为方式方法；into several parts 说明是分开、分裂，故选 spliting。

(4) 此题根据全文的主题词 spread spectrum(扩频)可知，与 frequencies(频率)有关。整句译为：这些设备利用 DSSS 技术彼此间通信，将数据分组，然后将它们用不同频率发送出去。

(5) bandwidth(带宽)，也和主题词频率有关。下句 about 22 megaherts(MHz)也说明了选项是描述频率范围的名词。

参 考 文 献

[1] 全国计算机技术与软件专业技术资格(水平)考试办公室编．软件设计师教程[M]．3 版．北京：清华大学出版社，2009．

[2] 全国计算机技术与软件专业技术资格(水平)考试办公室编．软件设计师考试同步辅导[M]．北京：清华大学出版社，2005．

[3] 全国计算机技术与软件专业技术资格(水平)考试办公室编．软件设计师考试大纲与培训指南(2009版)[M]．北京：清华大学出版社，2009．

[4] 全国计算机技术与软件专业技术资格(水平)考试办公室编．软件设计师历年试题分析与解答[M]．北京：清华大学出版社，2008．

[5] 胡圣明，褚华．软件设计师教程[M]．3 版．北京：清华大学出版社，2009．

[6] 王勇，唐强．软件设计师考试考点分析与真题详解(最新版)[M]．北京：电子工业出版社，2009．

[7] 郭春柱．软件设计师考试考前冲刺预测卷及考点解析(最新版)[M]．北京：电子工业出版社，2009．

[8] 严蔚敏，吴伟民．数据结构(C 语言版)[M]．北京：清华大学出版社，2007．

[9] [美]维斯．数据结构与算法分析(C 语言描述) [M]．北京：机械工业出版社，2004．

[10] 谢希仁．计算机网络[M]．5 版．北京：电子工业出版社，2008．

[11] [美]多切蒂．面向对象分析与设计(UML 2.0 版)[M]．北京：清华大学出版社，2006．

[12] 张海藩．软件工程导论[M]．5 版．北京：清华大学出版社，2006．

[13] [美]沙赫．软件工程面向对象和传统的方法[M]．北京：机械工业出版社，2007．

[14] 张文双，刘树明．C++培训教程[M]．北京：北京理工大学出版社，2007．

[15] 陈慧楠．算法设计与分析：C++语言描述[M]．北京：电子工业出版社，2006．

[16] 孙卫琴．Java 面向对象编程[M]．北京：电子工业出版社，2006．